

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Смоленский государственный университет»  
Кафедра прикладной математики и информатики

**«Утверждаю»**  
Проректор по учебно-  
методической работе  
Устименко Ю.А.  
«23» июня 2022 г.

**Рабочая программа дисциплины**  
**Б1.В.ДВ.07.02 Администрирование облачных технологий**

Направление подготовки **01.03.02 Прикладная математика и информатика**  
Направленность (профиль) **Математическое и информационное моделирование**  
Форма обучения: очная  
Курс – 4  
Семестр – 8  
Всего зачетных единиц – 2, часов – 72

Форма отчетности: зачет – 8 семестр

Программу разработал  
кандидат физико-математических наук, доцент Кристалинский В.Р.

Одобрена на заседании кафедры  
«16» июня 2022 г., протокол № 10

Заведующий кафедрой \_\_\_\_\_ С.В. Козлов

Смоленск  
2022

## 1. Место дисциплины в структуре ОП

Дисциплина «Администрирование облачных технологий» относится к дисциплинам части, формируемой участниками образовательных отношений (дисциплина по выбору). Она изучается на 4 курсе в 8 семестре. При ее изучении необходимы компетенции студентов, сформированные при изучении таких дисциплин, как «Основы информатики», «Операционные системы», «Структуры и алгоритмы компьютерной обработки данных», «Вычислительные системы, сети и телекоммуникации» и др.

В настоящее время облачные технологии широко применяются при разработке и использовании программного обеспечения, при обработке больших объемов данных. Поэтому компетенции, сформированные при изучении дисциплины, необходимы для написания выпускной квалификационной работы бакалавра и его дальнейшей профессиональной деятельности.

В связи с этим курс «Облачные технологии и программирование» занимает важное место в предметной подготовке бакалавров по направлению подготовки **01.03.02 Прикладная математика и информатика** (профиль «Математическое и информационное моделирование»).

Изучение курса основано на традиционных методах высшей школы, тесной взаимосвязи со смежными курсами, обобщающими методологию математического и информационного моделирования.

## 2. Планируемые результаты обучения по дисциплине

Компетенция	Индикаторы достижения
<b>ПК-1.</b> Способен осуществлять поиск, анализ, систематизацию научной информации в области прикладной математики и информатики для реализации научно-исследовательских проектов и решения прикладных задач по проектированию и разработке программного обеспечения.	<b>Знает:</b> теоретические основы и технологии организации научно-исследовательской деятельности. <b>Умеет:</b> осуществлять поиск, анализ, систематизацию научной информации в области прикладной математики и информатики для реализации научно-исследовательских проектов и решения прикладных задач по проектированию и разработке программного обеспечения. <b>Владеет:</b> навыками организации и проведения научно-исследовательской деятельности в ходе выполнения профессиональных функций.
<b>ПК-2.</b> Способен анализировать требования и проектировать программное и информационное обеспечение компьютерных сетей, вычислительные модели и модели данных для реализации элементов новых (или известных) программных продуктов.	<b>Знает:</b> возможности существующей программно-технической аппаратуры, современных и перспективных средств разработки программных продуктов, технических средств; методологии разработки программного обеспечения, технологии программирования; методы и средства проектирования программного обеспечения, баз данных, программных интерфейсов; принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения, типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; методики формализации и алгоритмизации поставленных задач. <b>Умеет:</b> проводить анализ требований к

	<p>программному обеспечению, вырабатывать варианты их реализации, проводить оценку и обоснование вырабатываемых решений; использовать существующие типовые решения и шаблоны проектирования программного обеспечения, применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов; использовать методы и приемы формализации и алгоритмизации задач, применять стандартные алгоритмы, использовать программные средства для графического отображения алгоритмов.</p> <p><b>Владеет:</b> методами анализа возможностей реализации требований к программному обеспечению, оценки времени и трудоемкости их реализации, навыками по проектированию программного обеспечения, баз данных, программных интерфейсов, информационных ресурсов сети Интернет.</p>
<p><b>ПК-3.</b> Способен разрабатывать и отлаживать программный код</p>	<p><b>Знает:</b> методологию разработки программного обеспечения, информационно-коммуникационных систем, баз данных, информационных ресурсов в сети Интернет; технологии программирования; особенности выбранной среды программирования и системы управления базами данных, синтаксис выбранного языка программирования, особенности программирования на нем, стандартные библиотеки языка программирования; компоненты программно-технических архитектур; методы повышения читаемости кода, системы кодировки символов, форматы хранения исходных текстов программ; методы и приемы отладки кода, типы и форматы сообщений об ошибках и состоянии аппаратных средств, современные компиляторы, отладчики оптимизаторы программного кода.</p> <p><b>Умеет:</b> применять выбранные языки программирования для написания программного кода, использовать выбранную среду программирования и средства системы управления базами данных, использовать возможности имеющейся технической и программной архитектуры; структурировать, комментировать, размечать, форматировать программный код в соответствии с требованиями; выявлять ошибки в программном коде, применять методы и приемы его отладки, интерпретировать сообщения об ошибках, предупреждениях,</p>

	<p>применять современные компиляторы, отладчики, оптимизаторы программного кода.</p> <p><b>Владеет:</b> навыками по созданию программного кода в соответствии с техническим заданием, оптимизации программного кода с использованием специализированных программных средств, форматированию программного кода, анализу, проверке, отладке исходного программного кода.</p>
--	--

### 3. Содержание дисциплины

**Тема 1. Тенденции развития современных инфраструктурных решений.** История основных типов высокопроизводительных вычислений, тенденции развития современных инфраструктурных решений. Знакомство с основными этапами развития вычислительной техники. Основные этапы развития аппаратного и программного обеспечения. Анализ современных тенденций развития аппаратного обеспечения, приведших к появлению технологий облачных вычислений. Базовые сведения о появлении, развитии и использовании технологий облачных вычислений. Основные современные тенденции развития аппаратного обеспечения, основные требования к инфраструктуре. Рост производительности компьютеров. Появление многопроцессорных и многоядерных вычислительных систем, развитие блейд-систем. Появление систем и сетей хранения данных. Консолидация инфраструктуры.

**Тема 2. Виртуализация. Сервисы. Основные направления развития.** Основные типы виртуализации. Обзор программных продуктов крупнейших компаний виртуализации. Виртуальная машина. Виртуализация серверов. Виртуализация приложений. Виртуализация представлений (рабочих мест). Разновидности архитектуры гипервизора.

**Тема 3. Введение в понятия облачных вычислений.** Обзор парадигмы облачных вычислений, Архитектура облачных систем. Модели развёртывания облаков: частное облако, публичное облако, гибридное облако, общественное облако. Основные модели предоставления услуг облачных вычислений: Software as a Service (SaaS) (ПО-как-услуга), Platform as a Service (PaaS), Инфраструктура как сервис (Infrastructure as a Service, IaaS), другие облачные сервисы (XaaS). Различия между облачными и кластерными (распределёнными, или – Grid-технологиями) вычислениями.

**Тема 4. Экономика облачных вычислений.** Достоинства и недостатки облачных вычислений. Основные преимущества и недостатки моделей облачных вычислений и предлагаемых на их основе решений. Экономика облачных вычислений. Термины и понятия.

**Тема 5. Обзор существующих сервисов. Обзор существующих платформ.** Обзор решений ведущих вендоров – Microsoft, Amazon, Google. Примеры облачных сервисов Microsoft. YandexCloud. Примеры облачных сервисов Google. Разработка и тестирование приложений на платформе Amazon Elastic Computing Cloud, Разработка облачных систем на платформе MapReduce, Разработка облачных систем на платформе Apache Hadoop.

**Тема 6. Технологии облачных вычислений.** Основные компоненты Cloud Computing: приложения, клиенты, инфраструктура, платформы, службы, хранение данных. Разработка Web-приложений для развёртывания в облачной среде, переноса в нее существующих приложений. Приемы программирования, навыки системного администрирования приложений, развёртываемых в облаке. Построение транзакционных Web-приложений, установка виртуальных серверов для их поддержки.

**Тема 7. Миграция из стандартной среды в облачные приложения.** Концепция миграции. Фазы миграции в облако. Выбор подходящей модели развёртывания в соответствии с существующими бизнес-задачами. Выбор подходящего поставщика облачных услуг.

### 4. Тематический план

№ п/п	Разделы и темы	Всего часов	Формы занятий			
			лекции	практические занятия	лабораторные занятия	самостоятельная работа
1	Тенденции развития современных инфраструктурных решений.	6	2	–	–	4
2	Виртуализация. Сервисы. Основные направления развития.	8	4	–	–	4
3	Введение в понятия облачных вычислений.	10	2	–	2	6
4	Экономика облачных вычислений.	6	2	–	–	4
5	Обзор существующих сервисов. Обзор существующих платформ. YandexCloud .	20	4	–	10	6
6	Технологии облачных вычислений.	16	4	–	8	4
7	Миграция из стандартной среды в облачные приложения.	6	2	–	–	4
<b>ИТОГО</b>		<b>72</b>	<b>20</b>	<b>–</b>	<b>20</b>	<b>32</b>

## 5. Виды образовательной деятельности

### Занятия лекционного типа

#### Тема 1 Тенденции развития современных инфраструктурных решений.

#### Лекция 1. Современные инфраструктурные решения и облачные технологии (2 часа)

##### *План*

1. История основных типов высокопроизводительных вычислений, тенденции развития современных инфраструктурных решений.
2. Анализ современных тенденций развития аппаратного обеспечения, приведших к появлению технологий облачных вычислений. Базовые сведения о появлении, развитии и использовании технологий облачных вычислений.
3. Основные современные тенденции развития аппаратного обеспечения, основные требования к инфраструктуре. Рост производительности компьютеров.
4. Появление систем и сетей хранения данных. Консолидация инфраструктуры.

#### Тема 2. Виртуализация. Сервисы. Основные направления развития.

#### Лекция 2. Технология виртуализации (2 часа)

##### *План*

1. Основные типы виртуализации.
2. Обзор программных продуктов крупнейших компаний виртуализации.
3. Виртуальная машина.

#### Лекция 3. Типы виртуализации

**(2 часа)**

*План*

1. Виртуализация серверов.
2. Виртуализация приложений.
3. Виртуализация представлений (рабочих мест).

**Тема 3. Введение в понятия облачных вычислений.**

**Лекция 4. Модели развёртывания облаков**

**(2 часа)**

*План*

1. Архитектура облачных систем.
2. Модели развёртывания облаков: частное облако, публичное облако, гибридное облако, общественное облако.
3. ПО как сервис (Software as a Service, SaaS).
4. Платформа как сервис (Platform as a Service, PaaS).
5. Инфраструктура как сервис (Infrastructure as a Service, IaaS).
6. Другие облачные сервисы (XaaS).

**Тема 4. Экономика облачных вычислений.**

**Лекция 5. Экономические аспекты применения облачных вычислений.**

**(2 часа)**

*План*

1. Термины и понятия.
2. Факторы, влияющие на экономическую эффективность применения облачных вычислений

**Тема 5. Обзор существующих сервисов. Обзор существующих платформ.**

**Лекция 6 YandexCloud.**

**(2 часа)**

*План*

1. Сервисы YandexCloud.
2. Работа с виртуальными машинами в YandexCloud.

**Лекция 7. Обзор решений Amazon и Apache.**

**(2 часа)**

*План*

1. Разработка и тестирование приложений на платформе Amazon Elastic Computing Cloud.
2. Разработка облачных систем на платформе Apache Hadoop.

**Тема 6. Технологии облачных вычислений.**

**Лекция 8. Основные компоненты облачных вычислений**

**(2 часа)**

*План*

1. Основные компоненты Cloud Computing: приложения, клиенты, инфраструктура, платформы, службы, хранение данных.
2. Разработка Web-приложений для развёртывания в облачной среде, переноса в нее существующих приложений.

**Лекция 9. Разработка облачных приложений**

**(2 часа)**

*План*

1. Приемы программирования облачных приложений
2. Системное администрирование приложений, развертываемых в облаке.
3. Построение транзакционных Web-приложений, установка виртуальных серверов для их поддержки.

## **Тема 7. Миграция из стандартной среды в облачные приложения.**

### **Лекция 10. Концепция миграции.**

**(2 часа)**

*План*

1. Фазы миграции в облако.
2. Выбор подходящей модели развертывания в соответствии с существующими бизнес-задачами.
3. Выбор подходящего поставщика облачных услуг.

### **Занятия семинарского типа**

При выполнении лабораторных работ студенты обучаются использовать возможности облачных сервисов известных производителей. Основным средством, используемым при проведении лабораторных работ, является подписка на облачную платформу YandexCloud.

### **Лабораторное занятие №1.**

Введение в понятия облачных вычислений.

*Цель работы:* ознакомиться с программными продуктами крупнейших программ виртуализации.

*Программное обеспечение и материалы:* актуальные версии Microsoft Windows, порталы Prezi.com, Google Office.

#### *Задания*

1. Открыть (запустить) любой обозреватель Интернет (браузер).
2. Ввести в строке адреса: <http://prezi.com> и открыть его.
3. На странице <http://prezi.com> для входа в систему нажмите Login.
4. Введите адрес электронной почты и пароль указанные при регистрации.
5. Перейдите на страницу Learn (Учиться). Здесь вы можете посмотреть обучающие видео ролики по работе с порталом. Просмотрите первый видео ролик по работе с порталом.
6. Перейдите на страницу Explore. На этой странице вы можете осуществлять поиск уже существующих презентаций находящихся в открытом доступе у других пользователей данной системы. Найдите все презентации о Photoshop. Посмотрите одну из презентаций на ваш выбор.
7. Перейдите на страницу Your prezis (Ваши презентации). На данной странице отображаются все ваши презентации.
8. Перейдите по ссылке New prezi.
9. В открывшемся окне Create a new Prezi (Создайте новую презентацию).
10. В поле Title (заголовок) введите следующий текст: «Моя первая презентация».
11. В поле Description (описание) введите следующее описание презентации «Это моя первая презентация на портале PREZI.COM». Примечание: Описание нужно для упрощения поиска вашей презентации в банке презентаций портала PREZI.COM другими пользователями, поэтому здесь лучше всего указывать краткое содержание презентации.
12. Нажмите Create new prezi (Создать новую презентацию).
13. На странице Templates (шаблоны) ознакомьтесь с различными шаблонами презентаций.
14. Выберите шаблон Blank (пустой) нажав на него.

15. Нажмите Chose (Выбрать).
16. Ознакомьтесь с инструментарием редактора презентаций портала PREZI.COM.
17. Заполните таблицу.
18. Увеличьте масштаб в 5 раз с помощью кнопки Zoom in (увеличение). Для этого подведите курсор манипулятора мышь к правой границе окна, появляется кнопки управления масштабом. Выберите Zoom in.
19. Уменьшите масштаб в 10 раз с помощью кнопки Zoom out (уменьшить). Для этого подведите курсор манипулятора мышь к правой границе окна, появляется кнопки управления масштабом. Выберите Zoom out.
20. Вернитесь к исходному значению с помощью иконки Show Overview (показать обзор). Так же изменить масштаб можно с помощью колесика манипулятора мышь.
21. Выйдите из редактора презентаций, нажав на иконку выход.
22. Нажмите на ваше имя и фамилию в верхнем правом углу окна портала PREZI.COM. В открывшемся меню выберите пункт Profile Page (Страница профиля) для перехода на страницу вашего профиля портала PREZI.COM.
23. Нажмите на ваше имя и фамилию в верхнем правом углу окна портала PREZI.COM. В открывшемся меню выберите пункт Settings & Account (Настройки и счета) для перехода в личный кабинет.
24. Уберите галочки с пункта Receive notifications if a comment is made on one of your prezis (Получать уведомления, если был оставлен комментарий на одной из ваших презентаций). Нажмите Save changes для сохранения внесенных вами изменений.
25. Уберите галочки с пункта Receive notifications if a reply is made to one of your comments (Получать уведомления, если был оставлен ответ на комментарий на одной из ваших презентаций). Нажмите Save changes для сохранения внесенных вами изменений.
26. В разделе Public profile link (Общественная ссылка профиля) укажите в качестве адреса сокращенное название вашего образовательного учреждения латинскими буквами, затем без укажите вашу фамилию и инициалы латинскими буквами. В качестве пробелов вы можете использовать знак нижнего подчеркивания. Нажмите Save changes для сохранения внесенных вами изменений.
27. В разделе Password change (Изменить пароль) при желании вы можете заменить пароль. Напомним, что для более усиленной защиты нужно менять пароль учетной записи раз в несколько месяцев, что делает возможность постороннего доступа личному кабинету почти невозможной.
28. Закройте сессию на портале PREZI.COM. Для этого нажмите на вашу фамилию и имя в правом верхнем углу страницы. В открывшемся меню выберите пункт Logout.
29. Закройте обозреватель Интернет (браузер).

1. Открыть (запустить) любой обозреватель Интернет (браузер).
2. Ввести в строке адреса: <http://prezi.com> и открыть его.
3. На странице <http://prezi.com> для входа в систему нажмите Login и введите адрес электронной почты и пароль, указанные при регистрации.
4. На странице Your Prezis (Ваши презентации) нажмите New Prezi (Новая презентация).
5. В открывшемся окне Create a new Prezi (Создайте новую презентацию).
6. В поле Title (заголовок) введите следующий текст: «Информация и её свойства».
7. В поле Description (Описание) придумайте и введите описание презентации.
8. Нажмите Create new prezi (Создать новую презентацию).
9. На странице Templates (шаблоны) выберите шаблон Blank (пустой) нажав на него.
10. Нажмите Chose (Выбрать).
11. Уменьшите масштаб в два раза.
12. Нажмите левую клавишу мыши в верхней части рабочей области. В этом месте появится текстовое поле.
13. Выберите стиль шрифта Title.



14. Установите красный цвет в качестве цвета текстового поля. Для этого нажмите на иконку цвета текста, и в появившемся списке цветов выберите красный цвет.
15. Введите заголовок презентации: «Информация и её свойства». 16. Нажмите левую клавишу манипулятора мышь на текст: «Click to add text» (Нажмите, чтобы добавить текст).
17. Выберите стиль шрифта Body для добавления основного текста.
18. Введите следующий текст:  
Информация – это сведения об окружающем нас мире. Информация – это продукт взаимодействия данных и методов их обработки, адекватных решаемой задаче.
19. Уменьшите размер текстового поля, до того момента пока текст не поместится в границу круга. Для этого перетащите круг в нижнем правом углу текстового поля.
20. Создайте текстовое поле в любом месте рабочей области вокруг определения понятия информации.
21. Выберите стиль шрифта Body для добавления основного текста.
22. Введите одно из свойств информации в созданное поле.
23. Нажмите на стрелку в правом верхнем углу текстового поля, для того чтобы раскрыть полную панель форматирования текстового поля редактора презентаций. Выберите пункт выравнивание, затем в отрывшемся окне выберите по ширине.
25. Повторите выполнение п.20-24 для всех свойств информации, расположив их вокруг определения понятия информации.
26. Добавьте в каждое текстовое поле содержащее свойство определение данного свойства.
27. Кликните левой клавишей мыши на определения понятия информации.
28. Выберите Edit Text (Изменить текст).
29. Из второго определения понятия информации создайте новое текстовое поле. Для этого установите курсор во второе определение понятия информации. Затем нажмите и перетащите значок, который появится в левой части текстового поля.
30. Установите синий цвет в качестве цвета текста текстового поля содержащего второе определение понятия информации.
31. Кликните левой клавишей мыши на определения понятия информации (Информация – это сведения об окружающем нас мире).
32. Выберите Delete (Удалить).
33. Поместите текстовое поле, содержащее в себе определение понятия информация, в центр круга.
34. Сохраните работу, нажав иконку Save, и выйдите из режима редактирования презентации, нажав иконку выход.
35. Закройте сессию на портале PREZI.COM. Для этого нажмите на вашу фамилию и имя в правом верхнем углу страницы. В открывшемся меню выберите пункт Logout.
36. Закройте обозреватель Интернет (браузер).

Создать на странице Google office свой аккаунт. Для этого достаточно иметь свой почтовый ящик и зарегистрироваться по адресу <https://docs.google.com/>. При регистрации вводится логин и почтовый ящик.

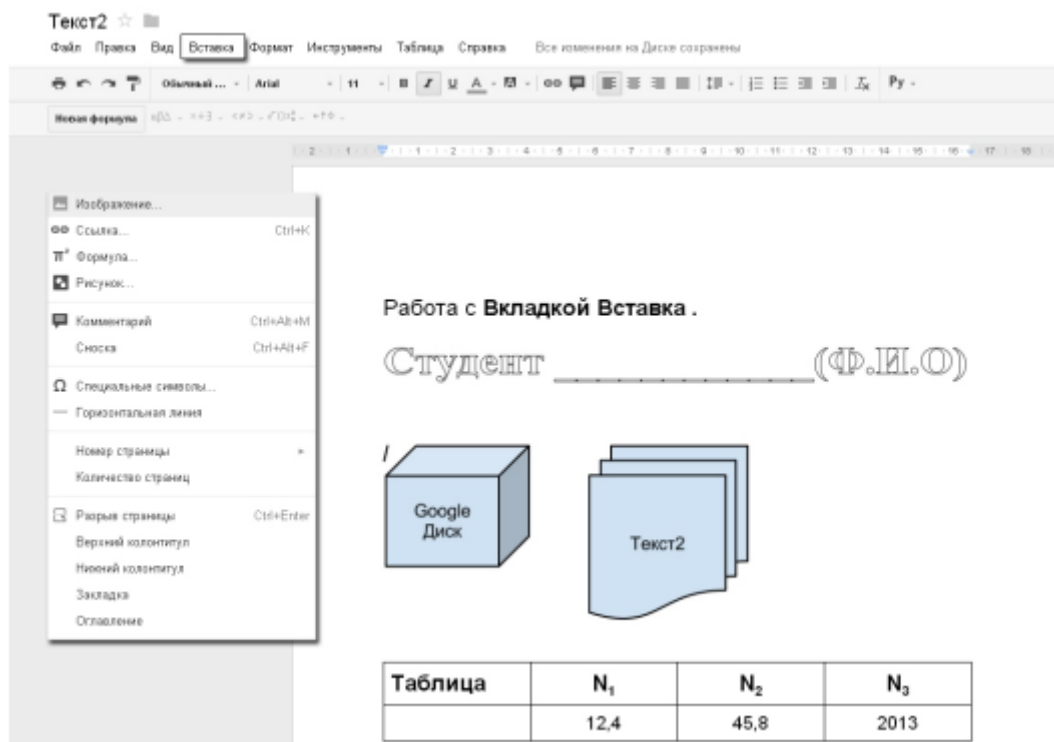
Документы Google - это пакет продуктов, предназначенный для создания различных документов, работы над ними вместе с другими пользователями в режиме реального времени и хранения документов и других файлов в Интернете.

После регистрации окно Диска кнопкой СОЗДАТЬ предлагает выбрать документ из стандартного списка офисных пакетов (Документ, Презентация, Таблица и т.д.). Создать Документ с именем «Текст1».

В окне редактора создаем текст и форматируем абзацы как на рисунке. Для точного выполнения задания использовать Вкладки Вид и Формат.

Функция автосохранения включена в сервисе автоматически. После выполнения задания убедитесь, что в списке Ваших документов появился текстовый документ «Текст1». В правой части окна находится информация о последнем изменении документа.

Создать Документ с именем «Текст2». В окне редактора применяя Вкладку Вставка, вставить объект wordArt. Два подписанных графических объекта «кубик» и «пачка документов» с надписями. При создании третьего объекта - таблицы обратить внимание на присутствие нижних индексов в названии столбцов.



### Лабораторное занятие №2-3.

*Знакомство с YandexCloud.*

*Цель работы:* освоить основные приемы работы с YandexCloud. Научиться создавать виртуальные машины средствами YandexCloud.

*Программное обеспечение и материалы:* актуальные версии Microsoft Windows, YandexCloud.

Создайте виртуальную машину Windows с помощью сервиса Compute Cloud в консоли управления Yandex.Cloud и подключитесь к ней.

Перед началом работы

1. Войдите в консоль управления.
2. Если у вас еще нет каталога, создайте его.

Создайте виртуальную машину

Чтобы создать виртуальную машину:

1. В консоли управления выберите каталог, в котором будет создана виртуальная машина.
2. В списке сервисов выберите **Compute Cloud**.
3. Нажмите кнопку **Создать VM**.
4. В блоке **Базовые параметры:**

Введите имя и описание VM. Требования к имени:

Длина — от 3 до 63 символов.

Может содержать строчные буквы латинского алфавита, цифры и дефисы.

Первый символ – буква. Последний символ – не дефис.

#### Примечание

Имя виртуальной машины используется для генерации имени FQDN единожды — при создании VM. Если для вас важно имя FQDN, учитывайте это и задавайте нужное имя VM при создании.

Выберите зону доступности, в которой будет находиться виртуальная машина.

(опционально) Выберите или создайте сервисный аккаунт. Использование сервисного аккаунта позволяет гибко настраивать права доступа к ресурсам.

5. В блоке **Образы из Cloud Marketplace**:

Выберите **Windows**.

Выберите версию и редакцию операционной системы.

6. (опционально) В блоке **Диски** настройте загрузочный диск:

Укажите нужный размер диска.

Выберите тип диска.

7. (опционально) Если вы хотите создать виртуальную машину из существующего диска, в блоке **Диски** добавьте диск.

8. В блоке **Вычислительные ресурсы**:

Выберите платформу.

Укажите гарантированную долю и необходимое количество vCPU, а также объем RAM.

При необходимости сделайте виртуальную машину прерываемой.

(опционально) Включите программно-ускоренную сеть.

9. В блоке **Сетевые настройки**:

Укажите идентификатор подсети или выберите облачную сеть из списка. Если сети нет, нажмите кнопку **Создать новую сеть** и создайте ее:

В открывшемся окне укажите имя новой сети и выберите, к какой подсети необходимо подключить виртуальную машину. У каждой сети должна быть как минимум одна подсеть (если подсети нет, создайте ее). Затем нажмите кнопку **Создать**.

В поле **Публичный адрес** выберите способ назначения адреса:

**Автоматически** — чтобы назначить случайный IP-адрес из пула адресов Yandex.Cloud.

**Список** — чтобы выбрать публичный IP-адрес из списка зарезервированных заранее статических адресов. Подробнее читайте в разделе Сделайте динамический публичный IP-адрес статическим.

**Без адреса** — чтобы не назначать публичный IP-адрес.

(опционально) Создайте запись для VM в зоне DNS. Разверните блок **Настройки DNS для внутренних адресов** и укажите зону, FQDN и время жизни записи. Подробнее см. Интеграция Cloud DNS с Compute Cloud.

(опционально) Выберите опцию защиты от DDoS-атак.

Выберите подходящие группы безопасности (если соответствующего поля нет, для виртуальной машины будет разрешен любой входящий и исходящий трафик).

10. При создании виртуальной машины в операционной системе будет автоматически создан пользователь Administrator. В поле **Пароль** задайте пароль для этого пользователя, с которым можно будет войти на виртуальную машину по RDP.

Не используйте легко подбираемые пароли. Пароли должны соответствовать требованиям сложности политики безопасности Windows.

В образах Windows Server от Yandex.Cloud у пароля пользователя Administrator по умолчанию не истекает срок действия.

11. Нажмите кнопку **Создать VM**.

Виртуальная машина появится в списке. При создании виртуальной машине назначаются IP-адрес и имя хоста (FQDN). Эти данные можно использовать для доступа к виртуальной машине по RDP.

Подключитесь к виртуальной машине

В образах всех версий и редакций операционной системы Windows, подготовленных для запуска в Yandex.Cloud, включен Remote Desktop Protocol (RDP) с проверкой подлинности на уровне сети — Network Level Authentication (NLA). Перед подключением к виртуальной машине убедитесь, что NLA включен в настройках вашего компьютера. Когда VM будет запущена (в статусе RUNNING), вы сможете подключиться к ней по протоколу RDP.

Группы безопасности виртуальной машины должны разрешать входящий трафик по протоколу TCP на порт 3389.

#### **Примечание**

Группы безопасности находятся на стадии Preview. Если они недоступны в вашей сети, для виртуальной машины будет разрешен весь входящий и исходящий трафик, и дополнительной настройки не требуется.

Для подключения по протоколу RDP укажите публичный IP-адрес или FQDN VM. Доступ по FQDN возможен из другой VM Yandex.Cloud, если она подключена к той же сети. IP-адрес и FQDN можно узнать в консоли управления, в блоке **Сеть** на странице VM.

Для подключения к VM:

1. Нажмите **Пуск**.
2. В поле поиска введите **Подключение к удаленному рабочему столу** и выберите соответствующий пункт.
3. В окне **Подключение к удаленному рабочему столу** в поле **Компьютер** введите публичный IP-адрес VM, к которой необходимо подключиться.
4. Нажмите кнопку **Подключиться**.  
Укажите параметры учетной записи:  
**Имя пользователя** Administrator.  
**Пароль**, который вы задали при создании VM.
5. Нажмите кнопку **ОК**.

Подключитесь к удаленному рабочему столу. Для этого:

Включите удаленные подключения на компьютере, к которому требуется подключиться.

Убедитесь, что у вас установлена Windows 10 Pro. Чтобы это проверить, перейдите на **Пуск > Параметры > Система > О системе** и найдите **Выпуск**.

Когда все готово, выберите **Пуск > Параметры > Система > Удаленный рабочий стол** и включите параметр **Включить удаленный рабочий стол**.

Запомните имя компьютера. Оно понадобится позже.

Используйте удаленный рабочий стол для подключения к настроенному компьютеру.

В поле поиска на панели задач введите **Подключение к удаленному рабочему столу** и выберите **Подключение к удаленному рабочему столу**. В окне "Подключение к удаленному рабочему столу" введите имя компьютера, к которому необходимо подключиться (из шага 1), а затем нажмите кнопку **Подключиться**.

VM создается в одном из каталогов в вашем облаке и наследует права доступа этого каталога. Переместить VM в другой каталог невозможно.

У каждой VM есть идентификатор и имя. Идентификатор, уникальный в пределах Yandex.Cloud, генерируется автоматически при создании машины. Вы сами придумываете уникальное в пределах каталога имя VM и указываете его в первом блоке параметров новой VM.

## **Выбор образа / загрузочного диска**

При создании VM подключите загрузочный диск с операционной системой. Он обязателен: без него VM не будет работать.

Определите параметры загрузочного диска.

- Нажмите значок  $\updownarrow$ , чтобы выбрать **тип** диска: HDD или SSD.
- Передвигайте ползунок, чтобы установить **размер** диска в гигабайтах. Справа вы увидите сведения о производительности диска: максимальное количество операций чтения и записи, выполняемых в секунду (**Макс. IOPS**), максимальная пропускная способность (**Макс. bandwidth**). Производительность диска зависит от размера.

При создании VM сервис создаёт только загрузочный диск операционной системы. Но вы можете подключить дополнительные диски (не используются как загрузочные): пустые или восстановленные из снимка либо образа. Чтобы добавить ещё один диск, нажмите **Добавить диск**. В открывшемся окне задайте параметры и нажмите кнопку **Добавить**.

## Вычислительные ресурсы

При создании VM укажите, сколько ей требуется вычислительных ресурсов: количество и производительность ядер процессора (vCPU, т. е. виртуальный процессор), количество памяти (RAM). Если нагрузка изменится, вы можете остановить VM и изменить вычислительные ресурсы.

**Платформа** определяет тип физического процессора и набор допустимых конфигураций vCPU (virtual CPU) и RAM. Наведите указатель на значок ?, чтобы увидеть соответствие платформ и процессоров.

Укажите **гарантированную долю vCPU**, которая будет выделена VM, количество ядер и объем оперативной памяти. VM с гарантированной долей меньше 100% обеспечивают указанный уровень производительности с вероятностью временного повышения до 100%. Такие машины подходят для задач, которые не требуют постоянной гарантии 100%-й производительности vCPU.

**Дополнительно** вы можете сделать VM прерываемой, т. е. ее работа может быть принудительно остановлена для высвобождения ресурсов под обычные (не прерываемые) VM. Такие машины предоставляются с большой скидкой. Подробнее об этом вы можете узнать в дальнейших материалах курса.

## Сетевые настройки

При создании VM заполните настройки сетевого интерфейса: выберите подсеть, к которой подключится VM, настройте внутренний и публичный IP-адреса. Это позволит машине взаимодействовать с сервисами во внутренней сети и в интернете. Вы также можете выбрать группу безопасности для ограничения доступа к VM. Подробнее о настройке виртуальной

облачной сети и групп безопасности вы узнаете в теме 4.

Выберите **подсеть** облачной сети из выпадающего списка. Вы увидите подсети для зоны доступности, которую выбрали в блоке **Базовые параметры**.

Настройка сетевого интерфейса доступна только при создании VM. **IP-адрес** позволит машине взаимодействовать с другими сервисами во внутренней сети (и в интернете, если адрес публичный).

В поле **Публичный адрес** выберите способ назначения IP-адреса:

- **Автоматически** — в этом случае VM получит случайный публичный IP-адрес из пула адресов Yandex.Cloud. При перезапуске VM выданный адрес сохраняется, при остановке и повторном запуске выдается новый адрес.
- **Список** — позволяет выбрать публичный IP-адрес из списка ранее зарезервированных статических адресов с DDoS-защитой или без неё.
- **Без адреса** — VM будет дан только внутренний IP-адрес из пула адресов выбранной подсети.

## Доступ

Установите параметры безопасности. Создайте сервисный аккаунт для VM: он позволяет настраивать права доступа к ресурсам для приложений, выполняемых на машине.

Например, вы написали приложение, которое отслеживает статусы виртуальных машин. Для этого ей достаточно иметь права на просмотр, но программа работает от вашего имени, а у вас есть права на удаление виртуальных машин. Чтобы защититься от случайного удаления виртуальной машины вашей программой, вы можете создать сервисный аккаунт и дать ему доступ только на просмотр.

Для машин с ОС на основе Linux вы можете сразу создать учётную запись администратора системы с авторизацией по SSH-ключу. Для этого укажите логин и введите публичный SSH-

ключ.

Для машин на основе Windows в поле **Пароль** наберите пароль для пользователя Administrator

в соответствии с требованиями к сложности паролей этой операционной системы.

В поле **Дополнительно** вы можете разрешить доступ к серийной консоли VM. Серийная консоль — аналог консольного порта оборудования, через который пользователь получает доступ к VM вне зависимости от состояния сети или операционной системы. С помощью консоли можно устранять неисправности.

## Лабораторное занятие №4.

### *Получение доступ к серийной консоли*

*Цель работы:* Ознакомиться с технологиями облачных вычислений.

*Программное обеспечение и материалы:* актуальные версии MicrosoftWindows, YandexCloud.

#### *Задания*

Работа серийной консоли зависит от настроек операционной системы. Yandex Compute Cloud обеспечивает канал между пользователем и COM-портом виртуальной машины (VM) и не гарантирует, что консоль будет работать стабильно.

Для входа на VM с ОС **Windows** используйте подключение по протоколу RDP. С ним вы уже разобрались в одном из предыдущих практических занятий.

Вход на VM с ОС на базе **Linux** возможен двумя способами: по протоколу SSH с другого компьютера, либо через веб-интерфейс Yandex.Cloud.

В первом случае...

1. Для подключения к VM необходимо знать её идентификатор (ID). Перейдите в раздел Compute Cloud. По умолчанию откроется страница со списком VM. В крайнем правом столбце указан ID каждой VM.
2. Используйте ID и имя связанного с ним пользователя для входа. Вот шаблон команды подключения для Linux:

Скопировать код

```
ssh -t -p 9600 -o IdentitiesOnly=yes -i ~/.ssh/<имя закрытого ключа><ID виртуальной машины>.<имя пользователя>@serialssh.cloud.yandex.net
```

3. Вот так вы подключитесь к консоли, если в VM с ID fhm0b28lgfp4tkoa3jl6 есть пользователь yc-user:

Скопировать код

```
ssh -t -p 9600 -o IdentitiesOnly=yes -i ~/.ssh/id_rsa fhm0b28lgfp4tkoa3jl6.yc-user@serialssh.cloud.yandex.net
```

4. Чтобы отключиться от серийной консоли, нажмите клавишу Enter, а затем введите символы ~. (тильда и точка). В терминалах Linux для отключения также можно нажать Ctrl + D.

Чтобы авторизоваться через **веб-интерфейс**, вам сначала нужно задать пароль для пользователя, которого вы завели при создании VM. Повторите описанные выше шаги для входа по SSH, затем используйте утилиту passwd в привилегированном режиме — `sudo passwd <username>` — для установки пароля текущему пользователю: после ввода команду дважды наберите одинаковый пароль.

Затем в веб-консоли откройте страницу VM и через меню слева перейдите на страницу **Серийная консоль**. При авторизации используйте ранее созданный логин и только что добавленный пароль.

## Лабораторное занятие №5.

*Создание VM с 5% vCPU и использование мониторинга*

*Цель работы:* научиться создавать прерываемую VM.

*Программное обеспечение и материалы:* актуальные версии Microsoft Windows, Yandex Cloud.

*Задания*

1. На странице **Виртуальные машины** в разделе **Compute Cloud** начните создавать VM. Заполните имя и описание, выберите операционную систему CentOS 8.
2. В разделе **Вычислительные ресурсы** укажите гарантированную долю vCPU 5%. Другие параметры оставьте по умолчанию.
3. После создания и запуска VM в списке машин нажмите её название. Вы перейдёте на страницу VM. Затем на левой боковой панели выберите **Мониторинг**. Откроется страница, где в динамике показывается информация о загрузке процессора, операциях с диском и сетевой активности. По умолчанию видны данные за одни сутки.
4. Переключитесь на один час: вверху слева нажмите 1h (1 hour).
5. На графике видно, что при запуске использование процессорных ресурсов было высоким, а позже снизилось до приемлемого. Чтобы посмотреть точные значения в определённый момент — поместите указатель над линией графика. Вы увидите всплывающее окно с показателями для этой точки времени.
6. Теперь удалите VM. Для этого вернитесь в список VM, поставьте галочку слева от названия VM и на появившейся внизу контекстной панели нажмите **Остановить**.
7. Во всплывающем окне подтвердите действие и нажмите кнопку **Остановить**. Дождитесь смены статуса на **Stopped**.

8. Чтобы удалить VM, в списке VM справа напротив машины нажмите ... и в раскрывшемся меню выберите **Удалить**. Подтвердите действие.

Через некоторое время VM будет удалена.

## Лабораторное занятие №6.

### *Создание бакетов в YandexObjectStorage*

*Цель работы:* научиться создавать бакеты в YandexObjectStorage.

*Программное обеспечение и материалы:* актуальные версии Microsoft Windows, YandexCloud.

В консоли управления выберите каталог, в котором хотите создать бакет.

Нажмите кнопку **Создать ресурс** и выберите **Бакет**.

Введите имя бакета.

Имя бакета должно быть уникальным для всего Object Storage. Это имя используется как часть URL для доступа к данным и его будут видеть ваши пользователи.

При необходимости ограничьте максимальный размер бакета.

Размер 0 означает отсутствие ограничений и аналогичен включенной опции **Без ограничений**.

Чтобы загруженные файлы всегда были доступны извне Yandex.Cloud, выберите публичный тип доступа. Иначе для доступа к таким файлам необходимо будет создавать временную ссылку.

Выберите класс хранилища, который будет использоваться по умолчанию при загрузке объектов:

*Стандартное хранилище* предназначено для активной работы с объектами.

*Холодное хранилище* предназначено для длительного хранения объектов с редкими запросами на чтение.

Нажмите кнопку **Создать бакет** для завершения операции.

Загрузка файлов в бакет

Чтобы загрузить объект в бакет:

В консоли управления выберите каталог.

Выберите сервис **Object Storage**.

Нажмите на имя необходимого бакета.

Чтобы загрузить объекты в бакет, перетащите файлы на экран с бакетом или нажмите кнопку **Загрузить**.

Получение ссылки на скачивание файла

Чтобы получить ссылку на загруженный объект:

В консоли управления выберите каталог.

Выберите сервис **Object Storage**.

Нажмите на имя необходимого бакета.

Нажмите на имя объекта.

Нажмите кнопку **Получить ссылку**.

Если у вас бакет с ограниченным доступом:

В открывшемся окне укажите **Время жизни** ссылки в часах или днях. Максимальное время жизни ссылки — 7 дней.

Нажмите кнопку **Получить ссылку**.

Скопируйте полученную ссылку.

Полученной ссылкой вы можете поделиться или использовать ее в своем сервисе для доступа к файлу.

## Лабораторное занятие №7.

### *Создание баз данных в YandexDatabase*

*Цель работы:* научиться создавать бакеты в YandexDatabase.



## Создать базу данных

Чтобы создать базу данных Serverless:

1. В консоли управления выберите каталог, в котором будет создана база данных.
  2. В списке сервисов выберите **Yandex Database**.
  3. Нажмите кнопку **Создать базу данных**.
  4. Введите **Имя** базы. Требования к имени:
    - Длина — от 3 до 63 символов.
    - Может содержать строчные буквы латинского алфавита, цифры и дефисы.
    - Первый символ — буква. Последний символ — не дефис.
  5. В блоке **Тип базы данных** выберите опцию **Serverless**.
  6. Вам будут предложены значения по умолчанию для параметров создаваемой базы данных. Вы можете оставить их без изменений, они выбраны таким образом, чтобы вы могли эффективно начать работу. В будущем вы можете их изучить и поменять, если потребуется.
  7. Нажмите кнопку **Создать базу данных**.
- Дождитесь запуска базы данных. В процессе создания база будет иметь статус **Provisioning**, а когда станет готова к использованию — статус сменится на **Running**.

Чтобы создать YDB-таблицу

В консоли управления перейдите на страницу каталога и выберите сервис **Yandex Database**.

Выберите базу данных, в которой нужно создать таблицу.

Выберите **Создать** → **Таблица** в правой части страницы.

Настройте параметры таблицы:

**Имя** таблицы. Должно быть уникальным в рамках базы данных.

**Тип** таблицы — YDB-таблица.

Добавьте колонки:

**Имя** колонки. Должно быть уникальным в рамках таблицы.

**Тип** данных колонки. Независимо от типа данных, каждая колонка может содержать значение NULL.

**Первичный ключ**. Укажите, входит ли колонка в состав первичного ключа. По первичному ключу автоматически строятся первичные индексы.

При необходимости настройте вторичные индексы:

**Имя** вторичного индекса. Должно быть уникальным в рамках таблицы.

**Ключ** — одна или несколько колонок, которые образуют ключ для построения вторичного индекса.

Настройте политику партицирования:

**Нет** — таблица не партицируется.

**Uniform** — весь диапазон значений ключевой колонки типа Uint32 или Uint64 (от 0 до максимального значения) разбивается на интервалы одинаковой длины. При использовании такой политики задайте число интервалов в поле **Количество**.

**Explicit** — позволяет явно указать значения ключей, которые будут выступать границами для первоначального разбиения таблицы на партиции. Чтобы добавить очередное граничное значение, нажмите кнопку **Add split point**.

Настройте автопартицирование:

**By size** — если включено, то партиция разделяется на две при достижении определенного размера данных.

**By load** — если включено, то партиция разделяется на две если в течение некоторого промежутка времени испытывает высокую нагрузку (потребляет много процессорного времени).

Задайте расширенные настройки таблицы:

**Autopartition by size MBs** — пороговое значение размера данных, при котором срабатывает автопартиционирование по размеру.

**Min partitions** — количество партиций в таблице, ниже которого не происходит слияние партиций по размеру или нагрузке.

**Max partitions** — количество партиций в таблице, выше которого не происходит разбиение по размеру или нагрузке.

**Key bloom filter** — если включено, использует Блум-фильтр для поиска по ключу. В некоторых случаях позволяет ускорить чтение по ключу.

Нажмите кнопку **Создать таблицу**.

Чтобы отправить SQL-запрос к базе данных из консоли управления:

В консоли управления выберите каталог, в котором находится нужная база данных.

В списке сервисов выберите **Yandex Database**.

Выберите базу из списка.

Нажмите кнопку **SQL-запрос** и введите текст запроса. При составлении запроса вы можете пользоваться шаблонами:

Чтобы воспользоваться одним из стандартных шаблонов, выберите его из выпадающего списка справа от кнопки **SQL-запрос**.

Чтобы подставить в шаблон данные определенной таблицы, нажмите на значок в строке с нужной таблицей и выберите шаблон.

Нажмите кнопку **Выполнить**

### Лабораторное занятие №8-9.

#### *Решение задач классификации на виртуальных машинах YandexCloud*

*Цель работы:* научиться решать задачи классификации на виртуальных машинах YandexCloud.

1. Запустить созданную виртуальную машину YandexCloud.
2. Установить на ней MSVisualStudio.
3. Отладить на виртуальной машине и запустить с нее программы, реализующие алгоритмы классификации, преобразовав их из консольных приложений в приложения WindowsForms. Ожидаемые результаты исполнения программ представлены на рисунках.

```
// Read the Excel worksheet into a DataTable
```

```
DataTable table = new ExcelReader("examples.xls").GetWorksheet("Classification - Yin Yang");
```

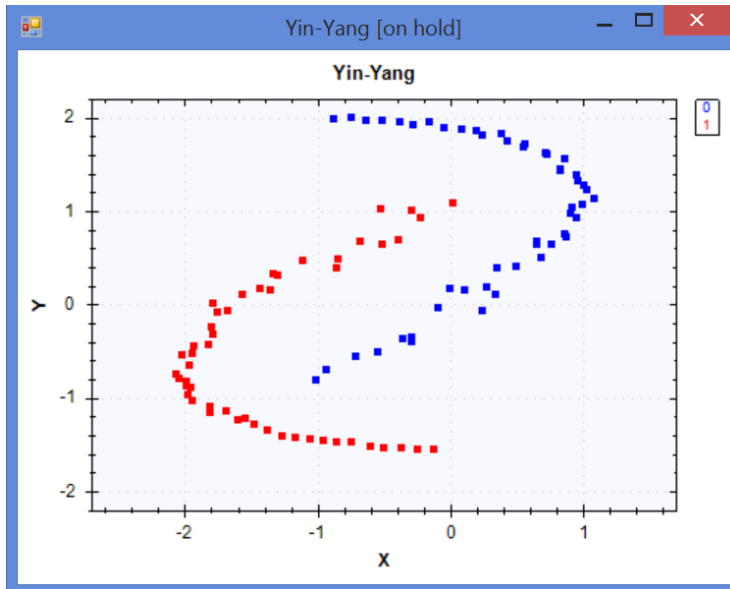
```
// Convert the DataTable to input and output vectors
```

```
double[][] inputs = table.ToJagged<double>("X", "Y");
```

```
int[] outputs = table.Columns["G"].ToArray<int>();
```

```
// Plot the data
```

```
ScatterplotBox.Show("Yin-Yang", inputs, outputs).Hold();
```



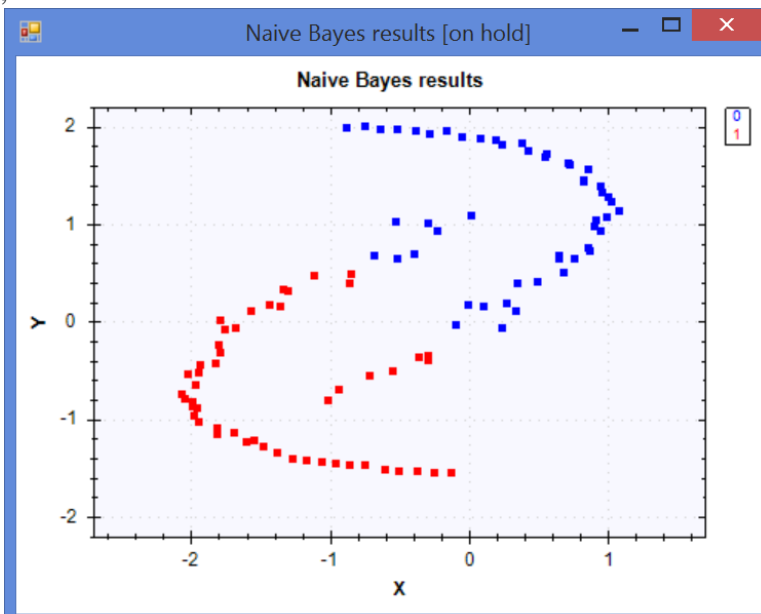
```
// Create a Naive Bayes learning algorithm
varteacher=newNaiveBayesLearning<NormalDistribution>();

// Use the learning algorithm to learn
varnb=teacher.Learn(inputs, outputs);

// At this point, the learning algorithm should have
// figured important details about the problem itself:
intnumberOfClasses=nb.NumberOfClasses; // should be 2 (positive or negative)
intnnumberOfInputs=nb.NumberOfInputs; // should be 2 (x and y coordinates)

// Classify the samples using the model
int[] answers=nb.Decide(inputs);

// Plot the results
ScatterplotBox.Show("Expected results", inputs, outputs);
ScatterplotBox.Show("Naive Bayes results", inputs, answers)
.Hold();
```



```

varteacher=newLinearCoordinateDescent();

// Teach the vector machine
varsvm=teacher.Learn(inputs, outputs);

// Classify the samples using the model
bool[] answers=svm.Decide(inputs);

// Convert to Int32 so we can plot:
int[] zeroOneAnswers=answers.ToZeroOne();

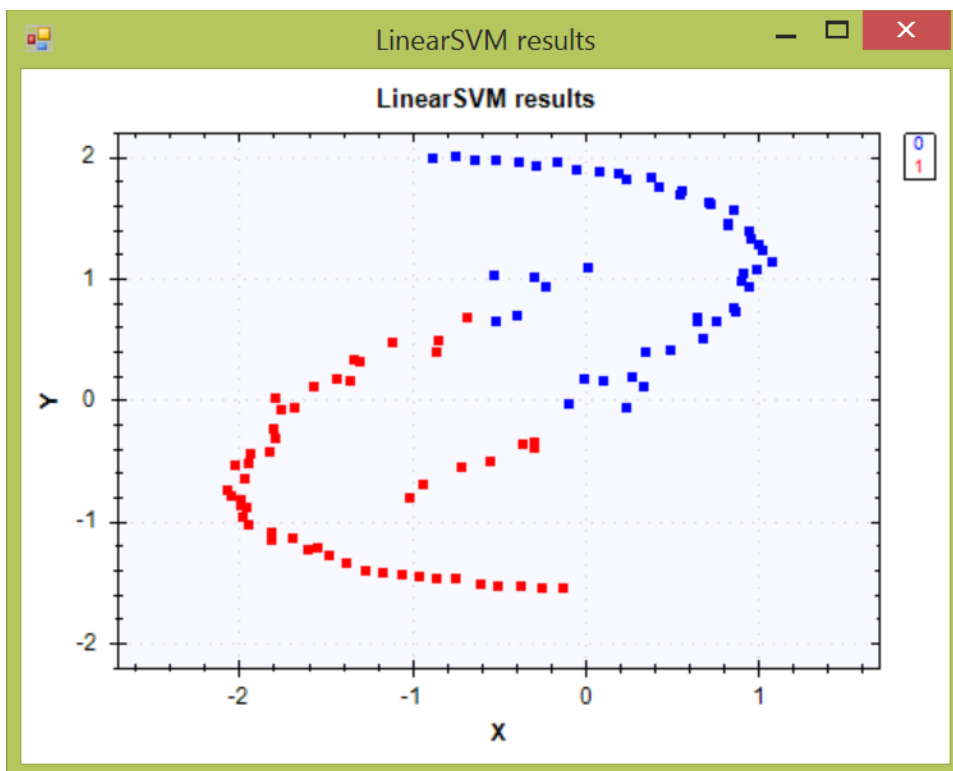
// Plot the results
ScatterplotBox.Show("Expected results", inputs, outputs);
ScatterplotBox.Show("LinearSVM results", inputs, zeroOneAnswers);

// Grab the index of multipliers higher than 0
int[] idx=teacher.Lagrange.Find(x=>x>0);

// Select the input vectors for those
double[][] sv=inputs.Get(idx);

// Plot the support vectors selected by the machine
ScatterplotBox.Show("Support vectors", sv).Hold();

```



### Лабораторное занятие №10.

*Решение задач регрессионного анализа на виртуальных машинах YandexCloud*

*Цель работы:* научиться решать задачи регрессионного анализа на виртуальных машинах YandexCloud.

1. Отладить на виртуальной машине и запустить с нее программы, реализующие алгоритмы регрессионного анализа, преобразовав их из консольных приложений в приложения WindowsForms.

### Простая регрессия

```
// Declare some sample test data.
double[] inputs= { 80, 60, 10, 20, 30 };
double[] outputs= { 20, 40, 30, 50, 60 };

// Use Ordinary Least Squares to learn the regression
OrdinaryLeastSquaresols=newOrdinaryLeastSquares();

// Use OLS to learn the simple linear regression
SimpleLinearRegressionregression=ols.Learn(inputs, outputs);

// Compute the output for a given input:
doubley=regression.Transform(85); // The answer will be 28.088

// We can also extract the slope and the intercept term
// for the line. Those will be -0.26 and 50.5, respectively.
doubles=regression.Slope; // -0.264706
doublec=regression.Intercept; // 50.588235
```

### Множественная регрессия

```
double[][] inputs=
{
// variables: x1 x2 x3
newdouble[] { 1, 1, 1 }, // input sample 1
newdouble[] { 2, 1, 1 }, // input sample 2
newdouble[] { 3, 1, 1 }, // input sample 3
};

double[][] outputs=
{
// variables: y1 y2
newdouble[] { 2, 3 }, // corresponding output to sample 1
newdouble[] { 4, 6 }, // corresponding output to sample 2
newdouble[] { 6, 9 }, // corresponding output to sample 3
};
// Use Ordinary Least Squares to create the regression
OrdinaryLeastSquaresols=newOrdinaryLeastSquares();

// Now, compute the multivariate linear regression:
MultivariateLinearRegressionregression=ols.Learn(inputs, outputs);

// We can obtain predictions using
double[][] predictions=regression.Transform(inputs);

// The prediction error is
doubleerror=newSquareLoss(outputs).Loss(predictions); // 0
```

```

// We will use Ordinary Least Squares to create a
// linear regression model with an intercept term
var ols = new OrdinaryLeastSquares()
{
    UseIntercept = true
};

// Now suppose you have some points
double[][] inputs =
{
    newdouble[] { 1, 1 },
    newdouble[] { 0, 1 },
    newdouble[] { 1, 0 },
    newdouble[] { 0, 0 },
};

// located in the same Z (z = 1)
double[] outputs = { 1, 1, 1, 1 };

// Use Ordinary Least Squares to estimate a regression model
MultipleLinearRegression regression = ols.Learn(inputs, outputs);

// As result, we will be given the following:
double a = regression.Coefficients[0]; // a = 0
double b = regression.Coefficients[1]; // b = 0
double c = regression.Intercept; // c = 1

// This is the plane described by the equation
//  $ax + by + c = z \Rightarrow 0x + 0y + 1 = z \Rightarrow 1 = z$ .

// We can compute the predicted points using
double[] predicted = regression.Transform(inputs);

// And the squared error loss using
double error = new SquareLoss(outputs).Loss(predicted);

```

### Логическая регрессия

```

double[][] input =
{
    // age, smokes?, had cancer?
    newdouble[] { 55, 0 }, // false - no cancer
    newdouble[] { 28, 0 }, // false
    newdouble[] { 65, 1 }, // false
    newdouble[] { 46, 0 }, // true - had cancer
    newdouble[] { 86, 1 }, // true
    newdouble[] { 56, 1 }, // true
    newdouble[] { 85, 0 }, // false
    newdouble[] { 33, 0 }, // false
    newdouble[] { 21, 1 }, // false
    newdouble[] { 42, 1 }, // true
};

```

```

bool[] output>// Whether each patient had lung cancer or not
{
false, false, false, true, true, true, false, false, false, true
};
// Create a new Iterative Reweighted Least Squares algorithm
varlearner=newIterativeReweightedLeastSquares<LogisticRegression>()
{
Tolerance=1e-4, // Let's set some convergence parameters
Iterations=100, // maximum number of iterations to perform
Regularization=0
};

// Now, we can use the learner to finally estimate our model:
LogisticRegressionregression=learner.Learn(input, output);
// For the age variable, we have that individuals with
// higher age have 1.021 greater odds of getting lung
// cancer controlling for cigarette smoking.
doubleageOdds=regression.GetOddsRatio(1); // 1.0208597028836701

doubleSmokeOdds=regression.GetOddsRatio(2); // 5.8584748789881331

// If we would like to use the model to predict a probability for
// each patient regarding whether they are at risk of cancer or not,
// we can use the Probability function:

double[] scores=regression.Probability(input);

// Finally, if we would like to arrive at a conclusion regarding
// each patient, we can use the Decide method, which will transform
// the probabilities (from 0 to 1) into actual true/false values:

bool[] actual=regression.Decide(input);

```

### **Самостоятельная работа**

Самостоятельная работа студентов направлена на углубление и закрепление знаний, а также развитие практических умений и заключается в:

- работе с лекционным материалом, поиске и анализе литературы и электронных источников информации;
- выполнении домашних заданий (домашние задания представляют из себя перечень задач, с которыми студенты не справились в ходе выполнения лабораторных работ);
- изучении теоретического материала к лабораторным занятиям.

Проверка качества самостоятельной работы студентов проводится во время защиты лабораторных работ. Студент должен ориентироваться в теоретической базе, необходимой для выполнения текущей работы, выполнить все задания, уметь отвечать на контрольные вопросы по направлению данной работы.

Консультирование студентов осуществляется в индивидуальном порядке на занятиях и во внеурочное время. Выполнение самостоятельной работы оценивается по электронным материалам, подготовленным студентами. Результаты деятельности накапливаются в индивидуальных портфолио студентов.

## 6. Критерии оценивания результатов освоения дисциплины (модуля)

### 6.1. Оценочные средства и критерии оценивания для текущей аттестации

#### Теоретические вопросы

1. Этапы развития IT инфраструктуры.
2. Современные инфраструктурные решения.
3. Создание нового проекта CloudService.
4. Модели виртуализации.
5. Преимущества виртуализации.
6. Виртуализация серверов.
7. Сетевые модели «облачных» сервисов.
8. Публичное «облако». Архитектуры публичных «облаков».
9. Преимущества и недостатки архитектуры публичного «облака».
10. Infrastructure-as-a-Service (IaaS).
11. Software-as-a-Service (SaaS).
12. Преимущества и риски, связанные с SaaS.
13. Область применения SaaS.
14. Platform-as-a-Service (PaaS).
15. Конфигурация проекта Yandex Cloud.
16. Платформа Yandex Cloud.
17. Компоненты Yandex Cloud.
18. Yandex Cloud Object Storage.
19. Yandex Cloud Database.
20. Работа с данными средствами YandexCloud

#### Критерии оценивания теоретических вопросов

Каждому студенту предлагается ответить на 5 произвольных теоретических вопросов. Ответ по каждому вопросу оценивается от 0 до 1 балла (в зависимости от содержательности ответа). Итоговая оценка по теме в разрезе теоретических вопросов складывается по формуле:

$$R = 2 + \frac{3}{5} \sum_{i=1}^5 Q_i,$$

где  $Q_i$  – баллы за ответ по каждому из вопросов.

#### Задания для лабораторных работ и задания для самостоятельной работы

Полный список типовых задач и заданий для самостоятельной работы представлен в материалах каждой лабораторной работы.

Задания для лабораторных и самостоятельной работ, образцы решений основных типовых задач практики также размещены в системе дистанционного обучения СмолГУ ([www.moodle.smolgu.ru](http://www.moodle.smolgu.ru)).

#### Критерии оценивания заданий из лабораторных работ и заданий для самостоятельной работы

Уровень выполнения	Оценка
Задание выполнено в полном объеме.	5 (отлично)
Задание выполнено в полном объеме с незначительными техническими ошибками.	4 (хорошо)
Задание выполнено не полностью.	3 (удовлетворительно)
Задание не выполнено.	2 (неудовлетворительно)

Оценка за выполнение заданий по лабораторной работе вычисляется как среднее арифметическое оценок за каждое задание по данной лабораторной работе.



## 6.2. Оценочные средства и критерии оценивания для промежуточной аттестации

### Зачетная контрольная работа

1. Самостоятельно построить и развернуть облачное приложение.
2. Создать блок-схему алгоритма программы.
3. Подготовить документацию по приложению..

### Критерии оценивания зачетной контрольной работы

#### 1. Нормы оценивания работы

№ п/п	Структурная часть контрольной работы	Количество баллов (*)
1	Правильно реализован каждый метод решения	1 балл
2	Анализ результатов	2 балла

(\*) Возможна градация в 0,25 балла.

#### 2. Шкала оценивания работы:

п/п	Оценка	Количество баллов
1	Отлично	4,75-5
2	Хорошо	3,75-4,5
3	Удовлетворительно	3-3,5
4	Неудовлетворительно	менее 3

### Критерии получения зачета

Зачет выставляется по результатам работы студента в течение семестра.

Для получения зачета студент должен:

- выполнить задания лабораторных работ на оценку не ниже «удовлетворительно»;
- выполнить задания для самостоятельной работы на оценку не ниже «удовлетворительно»;
- ответить на теоретические вопросы на оценку не ниже «удовлетворительно».

## 7. Перечень основной и дополнительной учебной литературы

### 7.1. Основная литература

1. Бабичев, С. Л. Распределенные системы: учебное пособие для вузов / С. Л. Бабичев, К. А. Коньков. — Москва: Издательство Юрайт, 2020. — 507 с. — (Высшее образование). — ISBN 978-5-534-11380-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/457005> (дата обращения: 17.09.2021).

### 7.2. Дополнительная литература

1. Сысолетин, Е. Г. Разработка интернет-приложений: учебное пособие для вузов / Е. Г. Сысолетин, С. Д. Ростунцев; под научной редакцией Л. Г. Доросинского. — Москва: Издательство Юрайт, 2021. — 90 с. — (Высшее образование). — ISBN 978-5-9916-9975-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/472200> (дата обращения: 19.09.2021).

### 7.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Система дистанционного обучения СмолГУ ([moodle.smolgu.ru](https://moodle.smolgu.ru)).
2. Национальный открытый университет ([intuit.ru](https://intuit.ru)).
3. Национальная платформа открытого образования ([opened.ru](https://opened.ru))
4. [Cloud.Yandex.ru](https://cloud.yandex.ru)

## **8. Материально-техническое обеспечение**

**Учебная аудитория для проведения занятий лекционного типа**, оснащенная стандартной учебной мебелью, интерактивной доской, мультимедиапроектором, ноутбуком и колонками.

**Учебная аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации** - компьютерная аудитория с выходом в Интернет.

**Помещение для самостоятельной работы** – компьютерный класс с доступом к сети «Интернет» и ЭИОС СмолГУ.

## **9. Программное обеспечение**

KasperskyEndpointSecurity для бизнеса Стандартный АО «Лаборатория Касперского», лицензия 1FB6-161215-133553-1-6231.

Microsoft Open License, лицензия 49463448 в составе: Microsoft Windows Professional 7 Russian; Microsoft Office 2010 Russian.

Поисковые системы сети Интернет.

Система программирования MS Visual Studio 19.

YandexCloud.

Стандартные браузеры.

ДОКУМЕНТ ПОДПИСАН  
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 03B6A3C600B7ADA9B742A1E041DE7D81B0  
Владелец: Артеменков Михаил Николаевич  
Действителен: с 04.10.2021 до 07.10.2022