

Министерство науки и высшего образования Российской Федерации



Калужский филиал  
федерального государственного бюджетного  
образовательного учреждения высшего образования

*«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»*

*(КФ МГТУ им. Н.Э. Баумана)*

**ФАКУЛЬТЕТ ИУК «Информатика и управление»**

**КАФЕДРА ИУК4 «Программное обеспечение ЭВМ, информационные технологии»**

**Ю.С. Белов, Г.Э. Амеличев**

**ТЕХНОЛОГИИ МАШИННОГО ОБУЧЕНИЯ В ОБЛАЧНОЙ  
СИСТЕМЕ YANDEX CLOUD**

*методические указания к лабораторной работе*

*по дисциплине*

***«Облачные технологии»***

*Калуга - 2023*

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	<b>Ошибка! Закладка не определена.</b>
ЦЕЛЬ И ЗАДАЧИ РАБОТЫ, ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ЕЕ ВЫПОЛНЕНИЯ .....	5
ЗАДАЧИ И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ .....	20
ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ .....	48
ТРЕБОВАНИЯ К РЕАЛИЗАЦИИ .....	49
КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ .....	49
ФОРМА ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ .....	50
ОСНОВНАЯ ЛИТЕРАТУРА .....	51
ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА .....	52

## **ВВЕДЕНИЕ**

Настоящие методические указания составлены в соответствии с программой проведения лабораторных работ по курсу «Облачные технологии» на кафедре «Программное обеспечение ЭВМ, информационные технологии» факультета информатики и управления Калужского филиала МГТУ им. Н.Э. Баумана.

Методические указания, ориентированные на студентов бакалавриата по направлению подготовки 09.03.04 «Программная инженерия», содержат краткую теоретическую часть, описывающую работу с облачным сервисом Yandex Cloud.

Методические указания составлены в расчете на всестороннее ознакомление студентов с основами работы с Yandex Cloud.

## **ЦЕЛЬ И ЗАДАЧИ РАБОТЫ, ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ЕЕ ВЫПОЛНЕНИЯ**

Целью выполнения лабораторной работы является приобретение практических навыков по работе с сервисом Yandex Cloud.

Основными задачами выполнения лабораторной работы являются:

- Ознакомиться с платформой Yandex Cloud.
- Изучить основы работы с Yandex Cloud.
- Научиться использовать возможности сервиса Yandex Cloud для задач машинного обучения.

Результатами работы являются:

- Разработанная программа согласно варианту задания
- Подготовленный отчет

## ХАРАКТЕРИСТИКА ОБЪЕКТА ИЗУЧЕНИЯ, ИССЛЕДОВАНИЯ

**Yandex Cloud** — публичная облачная платформа, которая предоставляет корпорациям, среднему бизнесу и частным разработчикам масштабируемую инфраструктуру, сервисы хранения данных, инструменты машинного обучения и средства разработки. С этими технологиями, проверенными временем и опытом Яндекса, каждый может создавать и непрерывно совершенствовать собственные ультрасовременные цифровые сервисы и приложения.

Yandex Cloud предлагает различные категории облачных ресурсов: например, виртуальные машины, диски, базы данных. Управлять ресурсами каждой категории можно с помощью соответствующего сервиса.

На данный момент в рамках платформы Yandex Cloud доступны следующие сервисы (и не только):

- Yandex API Gateway (Сервис для создания API-шлюзов).
- Yandex Application LoadBalancer (Сервис для распределения входящего трафика между компонентами веб-приложений).
- Yandex AuditTrails (Сервис сбора и выгрузки аудитных логов). Сервис находится на стадии Preview.
- Yandex Certificate Manager (Сервис для управления TLS-сертификатами).
- Yandex Cloud Backup (Сервис для резервного копирования ресурсов Yandex Cloud). Сервис находится на стадии Preview.
- Yandex Cloud CDN (Сервис для организации сети распространения контента).

Большая часть сервисов поддерживает использование меток для маркировки ресурсов. **Метка** — это пара ключ-значение в формате <имя

метки>=<значение метки>. Можно использовать метки для логического разделения ресурсов. На метку накладываются следующие ограничения:

- Максимальное количество меток на один ресурс: 64.
- Для ключа:
  - Длина — от 1 до 63 символов.
  - Может содержать строчные буквы латинского алфавита, цифры, дефисы и нижние подчеркивания.
  - Первый символ — буква.
- Для значения:
  - Длина — до 63 символов.

Может содержать строчные буквы латинского алфавита, цифры, дефисы и нижние подчеркивания.

Рассмотрим конкретно некоторые из сервисов и их возможности. Все ресурсы в Бессерверных вычислениях (Compute Cloud) можно разделить на 3 типа:

- Основные ресурсы — виртуальные машины и диски.
- Вспомогательные ресурсы — снимки и образы. Эти ресурсы используются только для создания других ресурсов.

Информационные ресурсы — тип диска и зона доступности. Эти ресурсы доступны только для чтения.

Помимо этого, при создании виртуальных машин используются ресурсы других сервисов: подсети и каталоги. Сервис Compute Cloud позволяет создавать виртуальные машины и подключать к ним диски. К виртуальной машине должен быть подключен как минимум один диск — загрузочный. Диск может быть восстановлен из снимка, из образа или создан пустым. Основное назначение снимков и образов — сохранение и восстановление дисков с данными. Снимок может быть создан только из диска. В информации о снимке содержится идентификатор диска, из которого снимок был создан. Образ может быть создан из диска, снимка, другого образа или из файла. При создании дисков и виртуальных машин

необходимо указать зону доступности, в которой они будут находиться (образы и снимки дисков не привязываются к зонам доступности). Можно посмотреть список зон доступности и узнать их текущий статус. При создании диска также указывается тип диска. Можно посмотреть возможные типы диска и узнать, в каких зонах они доступны.

**Виртуальная машина** — это аналог сервера в облачной инфраструктуре. ВМ создается в одном из каталогов в облаке и наследует права доступа от них. Каждая ВМ имеет уникальный идентификатор и имя. Имя уникально в рамках каталога. Идентификатор генерируется автоматически при создании ВМ и уникален в пределах Yandex Cloud. ВМ можно переместить в другой каталог внутри одного облака. При создании виртуальной машины можно выбрать, в какой из зон доступности Yandex Cloud она будет размещена. Переместить виртуальную машину в другую зону доступности невозможно, но возможно создать копию виртуальной машины в другой зоне доступности. При создании ВМ, указывается, сколько вычислительных ресурсов будет выделено машине: количество и производительность ядер процессора (vCPU), количество памяти (RAM). Можно выбрать подходящее количество вычислительных ресурсов из расчета планируемой нагрузки. К ВМ должен быть подключен как минимум один диск — загрузочный. На данный момент подключить загрузочный диск можно только при создании ВМ. Помимо этого к ВМ можно подключать дополнительные диски.

При создании виртуальные машины автоматически распределяются по физическому оборудованию Yandex Cloud. Чтобы контролировать уровень отказоустойчивости оборудования, виртуальные машины можно объединять в группы размещения.

**Группа размещения ВМ** — группа виртуальных машин, каждая из которых расположена на физическом оборудовании согласно определенной стратегии. Yandex Compute Cloud использует стратегию распределенного размещения.

**Распределенное размещение (spread)** — стратегия размещения виртуальных машин таким образом, чтобы каждая из виртуальных машин была гарантированно расположена на отдельной серверной стойке в одной из зон доступности. Если одна из стоек выйдет из строя, другие продолжат работу в обычном режиме.



Объединение ВМ в группу по стратегии распределенного размещения обеспечивает высокий уровень отказоустойчивости и снижает риск одновременного выхода из строя виртуальных машин, расположенных на одной серверной стойке. Однако из-за жестких требований к размещению вероятность нехватки физических ресурсов для виртуальных машин, объединенных в группу размещения, выше, в сравнении с тем же количеством машин, не объединенных в группу.

Виртуальная машина может находиться в одном из следующих статусов:

- **PROVISIONING** — Yandex Cloud выделяет ресурсы виртуальной машине.
- **STARTING** — запускается.
- **RUNNING** — запущена и работает.
- **STOPPING** — останавливается.
- **STOPPED** — остановлена.
- **RESTARTING** – перезагружается.
- **UPDATING** — обновляется.
- **CRASHED** — аварийно завершила работу и будет автоматически перезапущена.



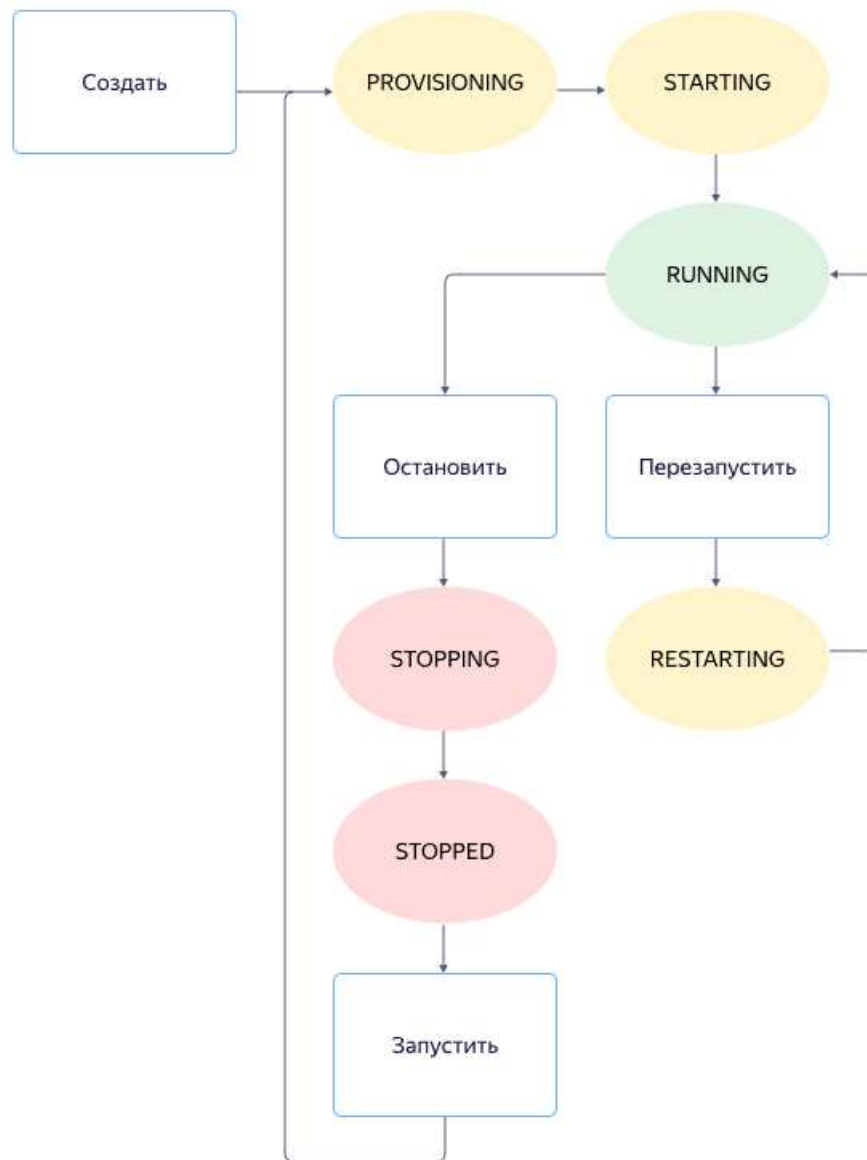
- ERROR — произошла фатальная ошибка и виртуальная машина не может быть восстановлена.
- DELETING — удаляется.

Диаграмма ниже отображает только переходы между статусами при штатной работе во время следующих действий:

- создание виртуальной машины;
- остановка и запуск виртуальной машины;
- перезапуск виртуальной машины.

На диаграмме не показаны следующие промежуточные статусы:

- UPDATING (обновление);
- DELETING (удаление);
- CRASHED и ERROR (возникновение ошибки).



При создании виртуальная машина получает статус PROVISIONING. На данном этапе машине выделяются вычислительные ресурсы: количество и производительность ядер процессора (vCPU), количество памяти (RAM). Назначается IP-адрес и создаются диски. После этого виртуальная машина автоматически запустится и перейдет в статус STARTING. Если запуск произошел успешно, машина перейдет в статус RUNNING и начнется загрузка операционной системы. После этого будет возможность подключиться к виртуальной машине.

При остановке виртуальная машина переходит в статус STOPPED. При следующем запуске виртуальная машина пройдет те же статусы, что и при создании: PROVISIONING → STARTING → RUNNING.

При перезапуске виртуальная машина переходит в статус RESTARTING. Так как все вычислительные ресурсы выделены, после перезагрузки машина вернется в статус RUNNING.

Для работы с данными к виртуальным машинам можно подключать следующие ресурсы Compute Cloud:

Ресурс	Описание
Диск	Виртуальный аналог физического накопителя, подключаемый к ВМ как сетевое блочное устройство. Один диск может быть подключен только к одной ВМ. С файловой системы диска можно снять копию — снимок диска.
Файловое хранилище	Файловые хранилища находятся на стадии Preview. Виртуальная файловая система, подключаемая к ВМ через интерфейс FilesysteminUserspace (FUSE) как устройство virtiofs. Одно файловое хранилище может быть подключено к нескольким ВМ одновременно.

К каждой ВМ можно подключить несколько дисков (помимо обязательного загрузочного) и несколько файловых хранилищ. Для дисков и файловых хранилищ действуют технические ограничения на операции чтения и записи.

**Диск** — это виртуальный аналог физических накопителей, таких как SSD и HDD. Диски предназначены для хранения данных и подключаются к виртуальным машинам для работы с ними. При отключении диска данные сохраняются. Каждый диск находится в одной из зон доступности

и реплицируется внутри нее (кроме нереплицируемых дисков), что обеспечивает сохранность данных. В другие зоны диски не реплицируются.

**Снимок диска** в Yandex Cloud — это копия файловой системы диска на определенный момент времени. Можно использовать снимки для различных целей, например:

- перенос данных с одного диска на другой — например, на диск в другой зоне доступности;
- создание резервной копии диска перед проведением операций, которые могут повлечь повреждение данных;
- версионирование диска путем регулярного создания снимков диска.

Если необходимо создавать много дисков с одинаковым содержимым, лучше использовать образы. Создание диска из образа происходит быстрее, чем из снимка. Снимок диска содержит только те данные, которые были записаны на диск в момент создания снимка. Если диск подключен к работающей виртуальной машине, то кеш приложений и операционной системы не попадет в снимок. Снимки дисков можно создавать вручную или автоматически, по расписаниям.

**Группа размещения нереплицируемых дисков** — группа нереплицируемых дисков, в которой диски располагаются в разных стойках в пределах одной зоны доступности. Если одна из серверных стоек с дисками группы выйдет из строя, это отразится на работе только одного диска. Объединение нереплицируемых дисков в группу снижает риск их одновременного выхода из строя. Однако из-за жестких требований к размещению вероятность нехватки физических ресурсов для нереплицируемых дисков, объединенных в группу размещения, выше, чем для того же количества дисков, не объединенных в группу. Возможно, создать нереплицируемый диск в группе размещения не удастся, хотя вне группы диск будет создан без проблем.

Файловое хранилище — это виртуальная файловая система, которую можно подключать к нескольким виртуальным машинам Compute Cloud из одной зоны доступности. С файлами в хранилище можно работать совместно, с нескольких ВМ. Каждое файловое хранилище находится в одной из зон доступности ru-central1-a или ru-central1-b и реплицируется внутри нее, что обеспечивает сохранность данных. Между зонами файловые хранилища не реплицируются.

Образ — это полная копия структуры файловой системы и данных, находящихся на диске. Образ предназначен для быстрого создания диска с данными, в первую очередь загрузочного диска виртуальной машины. Образ подходит для распространения программного обеспечения, например дистрибутивов операционных систем или дисков с уже установленными программами. При создании образа загрузочного диска с Windows нужно указать это в типе операционной системы. Это важно для правильной эмуляции процессора и для биллинга.

Instance Groups — компонент сервиса Compute Cloud, который позволяет создавать группы виртуальных машин и управлять ими. Instance Groups автоматически идентифицирует и корректирует неработоспособные виртуальные машины в группе для обеспечения их оптимальной работы. Каждая группа состоит из одной или нескольких однотипных виртуальных машин. Виртуальные машины группы могут находиться в разных зонах и регионах доступности.



С помощью Instance Groups можно:

- Одновременно обновлять все виртуальные машины в группе.
- Интегрироваться с сервисами Yandex Network LoadBalancer и Yandex Application LoadBalancer и равномерно распределять нагрузку между виртуальными машинами.
- Создавать автоматически масштабируемые группы виртуальных машин.
- Автоматически восстанавливать виртуальные машины в случае сбоя приложения.

Поддерживать работу служб приложений в надежной среде с многозональными функциями вместо выделения ресурсов для каждой зоны.

Выделенный хост — физический сервер, предназначенный для размещения исключительно пользовательских виртуальных машин в Yandex Cloud. VM на выделенных хостах обладают всеми возможностями обычных VM и дополнительно получают физическую изоляцию от VM других пользователей. Также можно разделить свои VM для разных проектов:



## ClickHouse

Облачные технологии YandexCloud имеют несколько вариантов решений. Рассмотрим один из сервисов на примере Базы Данных ClickHouse.

ClickHouse – столбцовая система управления базами данных (СУБД) для онлайн обработки аналитических запросов (OLAP). В обычной, «строковой» СУБД, данные хранятся в таком порядке:

Строка	WatchID	JavaEnable	Title	GoodEvent	EventTime
#0	89354350662	1	Investor Relations	1	2016-05-18 05:19:20
#1	90329509958	0	Contact us	1	2016-05-18 08:10:20
#2	89953706054	1	Mission	1	2016-05-18 07:38:00
#N	...	...	...	...	...

То есть, значения, относящиеся к одной строке, физически хранятся рядом. Примеры строковых СУБД: MySQL, Postgres, MS SQL Server. В столбцовых СУБД, данные хранятся в таком порядке:

Строка:	#0	#1	#2	#N
WatchID:	89354350662	90329509958	89953706054	...
JavaEnable:	1	0	1	...
Title:	Investor Relations	Contact us	Mission	...
GoodEvent:	1	1	1	...
EventTime:	2016-05-18 05:19:20	2016-05-18 08:10:20	2016-05-18 07:38:00	...

В примерах изображён только порядок расположения данных. То есть, значения из разных столбцов хранятся отдельно, а данные одного столбца - вместе. Примеры столбцовых СУБД: Vertica, Paracel (Actian Matrix, Amazon Redshift), Sybase IQ, Exasol, Infobright, InfiniDB, MonetDB (VectorWise, Actian Vector), LucidDB, SAP HANA, Google Dremel, Google PowerDrill, Druid, kdb+.

Разный порядок хранения данных лучше подходит для разных сценариев работы. Сценарий работы с данными – это то, какие производятся запросы, как часто и в каком соотношении; сколько читается данных на запросы каждого вида - строк, столбцов, байт; как соотносятся чтения и обновления данных; какой рабочий размер данных и насколько локально он используется; используются ли транзакции и с какой изолированностью; какие требования к дублированию данных и логической целостности; требования к задержкам на выполнение и пропускной способности запросов каждого вида и т. п.

Чем больше нагрузка на систему, тем более важной становится специализация под сценарий работы, и тем более конкретной становится эта специализация. Не существует системы, одинаково хорошо подходящей под существенно различные сценарии работы. Если система подходит под широкое множество сценариев работы, то при достаточно большой нагрузке,



система будет справляться со всеми сценариями работы плохо, или справляться хорошо только с одним из сценариев работы.

### **Ключевые особенности OLAP сценария работы:**

- подавляющее большинство запросов - на чтение;
- данные обновляются достаточно большими пачками (> 1000 строк), а не по одной строке, или не обновляются вообще;
- данные добавляются в БД, но не изменяются;
- при чтении, вынимается достаточно большое количество строк из БД, но только небольшое подмножество столбцов;
- таблицы являются «широкими», то есть, содержат большое количество столбцов;
- запросы идут сравнительно редко (обычно не более сотни в секунду на сервер);
- при выполнении простых запросов, допустимы задержки в районе 50 мс;
- значения в столбцах достаточно мелкие - числа и небольшие строки (пример - 60 байт на URL);
- требуется высокая пропускная способность при обработке одного запроса (до миллиардов строк в секунду на один сервер);
- транзакции отсутствуют;
- низкие требования к консистентности данных;
- в запросе одна большая таблица, все таблицы кроме одной маленькие;
- результат выполнения запроса существенно меньше исходных данных - то есть, данные фильтруются или агрегируются; результат выполнения помещается в оперативку на одном сервере.

Легко видеть, что OLAP сценарий работы существенно отличается от других распространённых сценариев работы (например, OLTP или Key-Value сценариев работы). Таким образом, не имеет никакого смысла пытаться

использовать OLTP или Key-Value БД для обработки аналитических запросов, если нужно получить приличную производительность. Например, если попытаться использовать для аналитики MongoDB или Redis – получится достаточно низкая производительность по сравнению с OLAP-СУБД.

### **Причины, по которым столбцовые СУБД лучше подходят для OLAP сценария.**

Столбцовые СУБД лучше (от 100 раз по скорости обработки большинства запросов) подходят для OLAP сценария работы.

#### **По вводу-выводу.**

1. Для выполнения аналитического запроса, требуется прочитать небольшое количество столбцов таблицы. В столбцовой БД для этого можно читать только нужные данные. Например, если требуется только 5 столбцов из 100, то следует рассчитывать на 20-кратное уменьшение ввода-вывода.
2. Так как данные читаются пачками, то их проще сжимать. Данные, лежащие по столбцам, также лучше сжимаются. За счёт этого, дополнительно уменьшается объём ввода-вывода.
3. За счёт уменьшения ввода-вывода, больше данных влезает в системный кэш.

Например, для запроса «посчитать количество записей для каждой рекламной системы», требуется прочитать один столбец «идентификатор рекламной системы», который занимает 1 байт в несжатом виде. Если большинство переходов было не с рекламных систем, то можно рассчитывать хотя бы на десятикратное сжатие этого столбца. При использовании быстрого алгоритма сжатия, возможно разжатие данных со скоростью более нескольких гигабайт несжатых данных в секунду. То есть, такой запрос может выполняться со скоростью около нескольких миллиардов строк в

секунду на одном сервере. На практике, такая скорость действительно достигается.

### **По вычислениям.**

Так как для выполнения запроса надо обработать достаточно большое количество строк, становится актуальным диспетчеризовывать все операции не для отдельных строк, а для целых векторов, или реализовать движок выполнения запроса так, чтобы издержки на диспетчеризацию были примерно нулевыми. Если этого не делать, то при любой не слишком плохой дисковой подсистеме, интерпретатор запроса неизбежно упрётся в CPU. Имеет смысл не только хранить данные по столбцам, но и обрабатывать их, по возможности, тоже по столбцам.

Есть два способа это сделать:

1. Векторный движок. Все операции пишутся не для отдельных значений, а для векторов. То есть, вызывать операции надо достаточно редко, и издержки на диспетчеризацию становятся пренебрежимо маленькими. Код операции содержит в себе хорошо оптимизированный внутренний цикл.
2. Кодогенерация. Для запроса генерируется код, в котором подставлены все косвенные вызовы.

В «обычных» БД этого не делается, так как не имеет смысла при выполнении простых запросов. Хотя есть исключения. Например, в MemSQL кодогенерация используется для уменьшения latency при выполнении SQL запросов. Для сравнения, в аналитических СУБД требуется оптимизация throughput, а не latency.

Стоит заметить, что для эффективности по CPU требуется, чтобы язык запросов был декларативным (SQL, MDX) или хотя бы векторным (J, K). То есть, чтобы запрос содержал циклы только в неявном виде, открывая возможности для оптимизации.

## ЗАДАЧИ И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

### Задание 1. Создание аккаунта

Итак, для начала работы с YandexCloud необходимо перейти на сайт <https://cloud.yandex.ru/>. После чего требуется нажать кнопку «Подключиться» в правом верхнем углу экрана, войти, используя либо YandexId, либо SSO. После входа требуется перейти по ссылке <https://console.cloud.yandex.ru/billing?section=accounts> и создать платёжный аккаунт, нажав в правом верхнем углу на кнопку «Создать аккаунт». После чего необходимо будет заполнить соответствующую форму и нажать кнопку «Создать».

Новым клиентам Yandex Cloud доступен стартовый грант. Стартовый грант может быть выдан физическому или юридическому лицу только один раз, если ранее вы:

- Никогда не приобретали услуг Yandex Cloud.
- Не активировали пробный период.

Стартовый грант действует 60 дней. Размер и валюта гранта зависят от страны, в которой вы проживаете:

- Для резидентов РФ размер гранта составляет не менее 4000 Р, с учетом НДС.
- Для резидентов РК размер гранта составляет не менее 24 000 ₸, с учетом НДС.
- Для нерезидентов РФ размер гранта составляет не менее 50 \$, без учета налогов и сборов.

Условия использования стартового гранта различаются для личных и бизнес-аккаунтов.

Личный аккаунт

Грант условно делится на две части и распределяется на сервисы следующим образом:

- *Первая часть* — 1000 Р от общего размера гранта. Может использоваться для оплаты услуг сервиса Yandex Compute Cloud, инструментов Yandex Cloud Marketplace и услуг технической поддержки.
- *Вторая часть* — 3000 Р от общего размера гранта. Может использоваться для оплаты услуг других сервисов Yandex Cloud.

Обе части могут использоваться одновременно.

### Гранты

Область действия	Баланс	Истекает через	Д
Все сервисы, кроме 3	3 000 Р / 3 000 Р	60 дней	1€
Compute Cloud +2	1 000 Р / 1 000 Р	60 дней	1€

Активировать промокод

### Гранты

Область действия	Баланс	Истекает через	Д
Все сервисы, кроме 3	3 000 Р / 3 000 Р	60 дней	1€
Compute Cloud +2	1 000 Р / 1 000 Р	60 дней	1€

Активировать промокод

0,00 Р

Область действия

Поиск по сервисам и продуктам

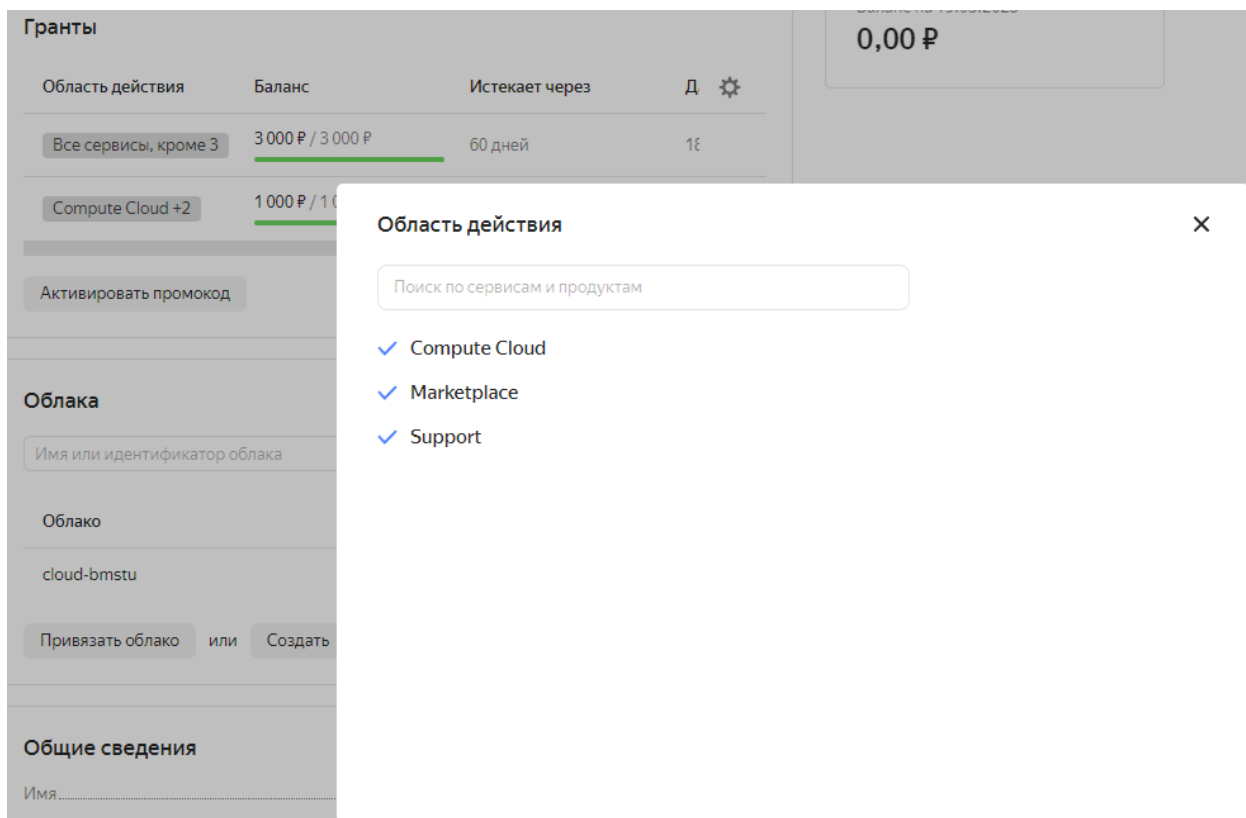
✓ Все сервисы

кроме 3 ^

✗ Compute Cloud

✗ Marketplace

✗ Support



## Пробный период

Пробный период позволяет вам получить практический опыт использования Yandex Cloud, не оплачивая потребленные ресурсы. В рамках пробного периода вашему платежному аккаунту выдается грант сроком действия 60 дней. Условия использования гранта различаются для личных и бизнес-аккаунтов. Вы можете перейти на платную версию Yandex Cloud с сохранением данных в любой момент во время пробного периода или в течение 60 дней после его завершения.

## Задание 2. Создание IAM токена

Для работы с API многих сервисов понадобится создать IAM токен. Инструкция по его созданию находится ниже. Важно помнить, что данный токен после создания жизнеспособен 12ч.

## Установка инструмента командной строки

Если у вас еще нет интерфейса командной строки Yandex Cloud, установите и инициализируйте его. Для Windows CLI можно установить с помощью PowerShell или командной строки:

***Для установки с помощью PowerShell:***

1) Выполните команду:

```
ieх (New-Object  
System.Net.WebClient).DownloadString('https://storage.yandexcloud.net/yandexcl  
oud-yc/install.ps1')
```

2) Скрипт установки спросит, нужно ли добавить путь до yc в переменную PATH:

Add yc installation dir to your PATH? [Y/n]

3) Введите Y. После этого CLI Yandex Cloud можно пользоваться, командную оболочку перезапускать не нужно.

***Для установки с помощью командной строки:***

1) Выполните команду:

```
@ "%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.exe" -  
Command "ieх ((New-Object  
System.Net.WebClient).DownloadString('https://storage.yandexcloud.net/yandexcl  
oud-yc/install.ps1'))" && SET "PATH=%PATH%;%USERPROFILE%\yandex-  
cloud\bin"
```

2) Скрипт установки спросит, нужно ли добавить путь до yc в переменную PATH:

Add yc installation dir to your PATH? [Y/n]

Введите Y.

Перезапустите командную оболочку, чтобы перезагрузить переменные окружения.

### Создание профиля

Чтобы аутентифицироваться от имени пользователя: Получите OAuth-токен в сервисе Яндекс.OAuth. Перейдите по ссылке: [https://oauth.yandex.ru/authorize?response\\_type=token&client\\_id=1a6990aa636648e9b2ef855fa7bec2fb](https://oauth.yandex.ru/authorize?response_type=token&client_id=1a6990aa636648e9b2ef855fa7bec2fb). Если приложение запрашивает доступ к данным, разрешите. Это необходимо для получения токена. Скопируйте в буфер обмена или сохраните полученный токен. Выполните команду `us init` для настройки вашего профиля CLI. По запросу команды введите свой OAuth токен.

1) Если вы выполняете команду `us init` не в первый раз, этот шаг будет отсутствовать.

Please go to

[https://oauth.yandex.ru/authorize?response\\_type=token&client\\_id=1a6990aa636648e9b2ef855fa7bec2fb](https://oauth.yandex.ru/authorize?response_type=token&client_id=1a6990aa636648e9b2ef855fa7bec2fb)

**in** order to obtain OAuth token.

Please enter OAuth token: AaAaBbBbCcCcDdDdEeEeFfFfGgGg

2) Выберите одно из предложенных облаков, к которым у вас есть доступ:

Please select cloud to use:

[1] cloud1 (id = aoe2bmdcvatao4frg22b)

[2] cloud2 (id = dcvatao4faoe2bmrg22b)

Please enter your numeric choice: 2

Если вам доступно только одно облако, оно будет выбрано автоматически.

Выберите каталог по умолчанию:



Please choose a folder to use:

[1] folder1 (id = cvatao4faoe2bmdrg22b)

[2] folder2 (id = tao4faoe2cvabmdrg22b)

[3] Create a new folder

Please enter your numeric choice: 1

3) Выберите зону доступности по умолчанию для сервиса Yandex Compute Cloud:

Do you want to configure a default Yandex Compute Cloud availability zone?

[Y/n] Y

Which zone do you want to use as a profile default?

[1] ru-central1-a

[2] ru-central1-b

[3] ru-central1-c

[4] Don't set default zone

Please enter your numeric choice: 2

4) Проверьте настройки вашего профиля CLI:

yc config list

Результат:

token: AaAaBbBbCcCcDdDdEeEeFfFfGgGg

cloud-id: b1g159pa15cddlv5mvcr

folder-id: b1g8o9jbt587mbadu25k

### **Генерация токена**

Для генерации токена вызовите команду `yc iam create-token`

### Задание 3. Click House

Сервис Managed Service for ClickHouse помогает разворачивать и поддерживать кластеры серверов ClickHouse в инфраструктуре Yandex Cloud. ClickHouse — это высокопроизводительная колоночная СУБД с открытым исходным кодом.

Managed Service for ClickHouse позволяет:

- Создавать системы, в которых требуется обработка и хранение большого объема данных.
- Обрабатывать аналитические запросы в режиме онлайн (OLAP).
- Поддерживать системы/работать с системами, которым нужна линейная масштабируемость по вертикали и горизонтали.
- Использовать параллельную обработку запросов на многих процессорных ядрах.

Взаимодействие с сервисом осуществляется с помощью консоли управления, интерфейса командной строки Yandex Cloud (CLI), API или HTTP-интерфейса. Managed Service for ClickHouse позволяет визуализировать структуру данных на вашем ClickHouse-кластере и отправлять SQL-запросы к базам из консоли управления Yandex Cloud.

*Перед началом работы:*

1. Перейдите в консоль управления, затем войдите в Yandex Cloud или зарегистрируйтесь, если вы еще не зарегистрированы.
2. Если у вас еще нет каталога, создайте его:
  1. В консоли управления выберите облако и нажмите кнопку **+ Создать каталог**.
  2. Введите имя каталога. 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.
3. (опционально) Введите описание каталога.
  4. Выберите опцию «Создать сеть по умолчанию». Будет создана сеть с подсетями в каждой зоне доступности. Также в этой сети будет создана группа безопасности по умолчанию, внутри которой весь сетевой трафик разрешен.
  5. Нажмите кнопку «Создать».
3. Подключаться к кластерам БД можно как изнутри, так и извне Yandex Cloud:
- Чтобы подключиться изнутри Yandex Cloud, создайте виртуальную машину на основе Linux в той же сети, что и кластер БД.
  - Чтобы подключиться к кластеру из интернета, запросите публичный доступ к хостам при создании кластера.

**Примечание:** Следующие шаги предполагают, что подключение к кластеру производится с ВМ на основе Linux.

Подключитесь к виртуальной машине по SSH.

Подключите DEB-репозиторий ClickHouse:

```
sudo apt update && sudo apt install -y apt-transport-https ca-certificates dirngr && \
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv E0C56BD4 && \
echo "deb https://repo.clickhouse.com/deb/stable/ main/" | sudo tee \
/etc/apt/sources.list.d/clickhouse.list
```

Установите зависимости и клиентское приложение clickhouse-client:

```
sudo apt update && sudo apt install -y clickhouse-client
```

Загрузите файл конфигурации для clickhouse-client:

```
mkdir -p ~/.clickhouse-client && wget
"https://storage.yandexcloud.net/mdb/clickhouse-client.conf.example"
-O ~/.clickhouse-client/config.xml
```

Для начала работы с сервисом требуется создать кластер, для этого:

1. В консоли управления выберите каталог, в котором нужно создать кластер БД.
2. Выберите сервис «Managed Service for ClickHouse».
3. Нажмите кнопку «Создать кластер».
4. Задайте параметры кластера и нажмите кнопку «Создать кластер».
5. Дождитесь, когда кластер будет готов к работе: его статус на панели Managed Service for ClickHouse сменится на «Running», а состояние — на «Alive». Это может занять некоторое время.

После чего требуется *подключиться к БД*:

1. Настройте группы безопасности для облачной сети так, чтобы был разрешен весь необходимый трафик между кластером и хостом, с которого выполняется подключение.
2. Для подключения к серверу БД получите SSL-сертификат:

```
sudo mkdir --parents /usr/local/share/ca-certificates/Yandex/ && \
sudo wget "https://storage.yandexcloud.net/cloud-certs/CA.pem" \
--output-document /usr/local/share/ca-certificates/Yandex/YandexCA.crt && \
sudo chmod 655 /usr/local/share/ca-certificates/Yandex/YandexCA.crt
```

Сертификат будет сохранен в файле /usr/local/share/ca-certificates/Yandex/YandexCA.crt.

3. Используйте для подключения ClickHouse CLI:

3.1. Укажите путь к SSL-сертификату в конфигурационном файле, в элементе <caConfig>:

```
<config>
  <openssl>
    <client>
      <loadDefaultCAFile>true</loadDefaultCAFile>
      <caConfig>/usr/local/share/ca-certificates/Yandex/YandexCA.crt</caConfig>
      <cacheSessions>true</cacheSessions>
      <disableProtocols>sslv2,sslv3</disableProtocols>
      <preferServerCiphers>true</preferServerCiphers>
      <invalidCertificateHandler>
        <name>RejectCertificateHandler</name>
      </invalidCertificateHandler>
    </client>
  </openssl>
</config>
```

### 3.2. Запустите ClickHouse CLI со следующими параметрами:

```
clickhouse-client --host <FQDN любого хоста ClickHouse> \  
--secure \  
--user <имя пользователя> \  
--database <имя БД> \  
--port 9440 \  
--ask-password
```

Для получения дополнительной информации следует воспользоваться ссылкой <https://cloud.yandex.ru/docs/managed-clickhouse/quickstart?from=int-console-empty-state> для более точечного взаимодействия с сервисом ClickHouse.

#### Задание 4. Yandex Speech

Голосовые технологии Yandex SpeechKit помогают решить любую задачу, связанную с речью человека. SpeechKit может распознавать речь в режиме реального времени и из предварительно записанных аудиофайлов, автоматически определяя язык говорящего, а также озвучивать шаблонные фразы и развернутые тексты при помощи стандартных голосов SpeechKit. SpeechKit работает через интерфейсы API. В зависимости от задачи вам доступны интерфейсы gRPC или REST. Подробнее о реализации API в Yandex Cloud см. Концепции API Yandex Cloud.

Примеры использования:

- Голосовой робот
- Речевая аналитика. Контроль качества работы операторов
- Голосовое управление приложением и умными устройствами
- Голосовой ассистент
- Адаптация сервиса для людей с нарушениями зрения
- Распознавание аудиозаписей встреч
- Озвучивание книг и видео
- Ведение протоколов совещаний

- Субтитры к видео
- Субтитры к трансляции
- Расшифровка голосовых сообщений

Преимущества использования Yandex speech:

- очень быстрое развертывание нужного рабочего окружения, независимого от других проектов
- относительная (с оговорками) дешевизна
- практически полное отсутствие необходимости поддержки серверной части (но лимиты на бюджет — обязательно)

Даже в платном режиме есть лимиты использования. За один запрос вы не можете отправить на озвучание более 5000 символов текста и не чаще чем 40 раз в секунду.

Пример показывает, как с помощью API v1 синтезировать речь из текста в TTS-разметке в файл формата WAV. В примере заданы следующие параметры синтеза:

- формат синтезированного аудиофайла — LPCM с частотой дискретизации 48 000 Гц, контейнер WAV;
- язык — русский;
- голос — jane.

Преобразование и запись результата в формат WAV выполняются с помощью утилиты SoX (<https://sox.sourceforge.net/>). Аутентификация происходит от имени аккаунта на Яндексе или федеративного аккаунта с помощью IAM-токена. Если вы используете сервисный аккаунт, передавать в запросе идентификатор каталога не нужно.

```
import os
import subprocess
import time
import requests

token = "Сюда записывается персональный токен"
folder_id = "id рабочего каталога в Yandex Cloud"

root_path = os.path.dirname(__file__)
```

```

def synthesize(text):
    url = 'https://tts.api.cloud.yandex.net/speech/v1/tts:synthesize'
    headers = {'Authorization': 'Bearer ' + token, }

    data = {
        'folderId': folder_id,
        'text': text,
        'lang': 'ru-RU',
        'voice': 'jane', # голос
        'emotion': 'evil', # эмоциональная окраска
        'speed': '1.1', # скорость
        'format': 'lpcm', # по умолчанию конвертит в oggopus, кот никто не
понимает, зато занимает мало места
        'sampleRateHertz': 48000,
    }

    with requests.post(url, headers=headers, data=data, stream=True) as resp:
        if resp.status_code != 200:
            raise RuntimeError("Invalid response received: code: %d, message:
%s" % (resp.status_code, resp.text))

        for chunk in resp.iter_content(chunk_size=None):
            yield chunk

def write_file(text):
    filename = str(int(time.time()))
    with open(filename + ".raw", "wb") as f:
        for audio_content in synthesize(text):
            f.write(audio_content)

    time.sleep(2)

    return filename

def convert(filename):
    # сборка команды конвертации
    cmd = " ".join([
        root_path + "\sox\sox.exe",
        "-r 48000 -b 16 -e signed-integer -c 1",
        filename + ".raw",
        filename + ".wav",
    ])

    # команда конвертации
    subprocess.Popen(cmd, stdout=subprocess.PIPE, stderr=subprocess.STDOUT,
universal_newlines=True)

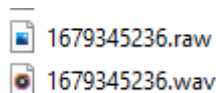
def read_text():
    with open("text.txt", "r", encoding="UTF-8") as f:
        text = f.read()

    return text

convert(write_file(read_text()))

```

В результате работы программы получаем .wav файл с озвучкой:



Будьте аккуратны при выборе голосов! Премиум голоса могут очень быстро исчерпать ваш пробный бюджет.

## Задание 5. Yandex Translate

Сервис для интеграции алгоритмов Яндекс Переводчика в приложения или веб-проекты для конечных пользователей. Yandex Translate поддерживает более 90 языков и умеет переводить отдельные слова и целые тексты. Для доступа к Translate используйте API.

### Yandex Translate

Сервис машинного перевода для интеграции в приложения и веб-сервисы.

Построен на основе опыта создания Яндекс Переводчика. Использует технологии нейронного машинного обучения.

Подключиться

Получить предложение

#### Поддержка 90+ языков

Yandex Translate умеет переводить отдельные слова и целые тексты в большом количестве языковых пар. Если вы не знаете, на каком языке написан текст, наш API определит язык исходного запроса.

#### Совершенствование алгоритмов

Алгоритмы непрерывно обучаются на новых текстах и расширяют базы перевода. Ваши данные, передаваемые в сервис, для этих целей не используются.

#### Гибридная модель перевода

Алгоритм использует две модели перевода — статистическую и нейронную. Такой подход позволяет с точностью переводить отдельные фразы и редкие слова, а также анализировать контекст в целом.

#### Собственные глоссарии

Вы можете создавать собственные глоссарии. Yandex Translate будет переводить специализированные тексты с учётом вашей терминологии.

В этом примере показано, как перевести на русский язык текст из заданного файла. Язык исходного текста будет определен автоматически.

```
import requests

token = "Сюда записывается персональный токен"
folder_id = "id рабочего каталога в Yandex Cloud"
target_language = 'ru'

with open("text.txt", "r", encoding="UTF-8") as f:
    text = f.read()

body = {
    "targetLanguageCode": target_language,
```



```

        "texts": text,
        "folderId": folder_id,
    }

headers = {
    "Content-Type": "application/json",
    "Authorization": "Bearer {0}".format(token)
}

response =
requests.post('https://translate.api.cloud.yandex.net/translate/v2/translate'
,
    json = body,
    headers = headers
)

print(response.text)

```

В результате работы программы получаем переведенное описание:

```

{
  "translations": [
    {
      "text": "Факультет отличается высоким профессиональным уровнем профессорско-преподавательского состава. В
      "detectedLanguageCode": "ru"
    }
  ]
}

```

## Задание 6. Yandex Vision

Yandex Vision — сервис компьютерного зрения для анализа изображений.

### Yandex Vision

Сервис компьютерного зрения для анализа изображений.

[Подключиться](#)

[Документация](#)



#### Поддержка 40+ языков

Сервис работает с разными языковыми моделями. В их числе латинско-кириллическая модель с поддержкой более 40 языков.

#### Простая интеграция

Сервис предоставляет REST и gRPC API, которые можно использовать для интеграции с вашими приложениями.

#### Высокая скорость обработки

Одна страница текста распознается за пару секунд. В одном запросе вы можете отправить сразу несколько изображений с текстами на одном или нескольких языках.

#### Классификация изображений [Preview](#)

Технология классификации изображений помогает анализировать контент, который загружают ваши пользователи. Например, вы можете фильтровать контент для взрослых или отбирать только изображения высокого качества.

#### Распознавание текста

Vision распознаёт текст на изображении, указывает его расположение и возвращает в виде блоков текста, строк и слов с автоматической расстановкой знаков препинания.

#### Обнаружение лиц [Preview](#)

Обнаружение лиц автоматически находит фотографии с портретами среди всех изображений, помогает отметить людей на фотографии.

#### Распознавание автомобильных номеров

Сервис распознаёт автомобильные номера на фотографиях, а также шаблоны документов (например, паспорта).

## Распознавание текста

Чтобы распознать текст с изображения или из PDF-файла, воспользуйтесь возможностью Распознавания текста. Для этого в методе `batchAnalyze` в свойстве `type` укажите `TEXT_DETECTION`, а в свойстве `textDetectionConfig` задайте настройки распознавания.

1) Подготовьте файл изображения, соответствующий требованиям:

- Поддерживаемые форматы файлов: JPEG, PNG, PDF.
- MIME-тип файла вы указываете в свойстве `mime_type`. По умолчанию `image`.
- Максимальный размер файла: 1 МБ.
- Размер изображения не должен превышать 20 мегапикселей (длина x ширина).

2) Кодировать файл в формат Base64:

```
import base64

def encode_file(file):
    file_content = file.read()
    return base64.b64encode(file_content)
```

3) Создайте файл с телом запроса, например `body.json`. В свойстве `content` укажите изображение, кодированное в Base64.

Чтобы сервис автоматически определил язык текста, укажите в конфигурации свойство `"language_codes": ["*"]`.

```
body.json:
{
  "folderId": "b1gvmob95yysaplct532",
  "analyze_specs": [{
    "content": "iVBORw0KGgo...",
    "features": [{
      "type": "TEXT_DETECTION",
      "text_detection_config": {
        "language_codes": ["*"]
      }
    }]
  }]
}
```

4) Отправьте запрос с помощью метода `batchAnalyze` и сохраните ответ в файл, например `output.json`:

```
export IAM_TOKEN=CggaATEVAgA...
```

```
curl -X POST \
-H "Content-Type: application/json" \
-H "Authorization: Bearer ${IAM_TOKEN}" \
-d '@body.json' \
https://vision.api.cloud.yandex.net/vision/v1/batchAnalyze > output.json
```

Ответ будет состоять из распознанных блоков текста, строк и слов с указанием их местоположения на изображении:

```
{
  "results": [{
    "results": [{
      "textDetection": {
        "pages": [{
          "blocks": [{
            "boundingBox": { ... },
            "lines": [{
              "boundingBox": { ... },
              "words": [{
                "boundingBox": {
                  "vertices": [{
                    "x": "504",
                    "y": "760"
                  },
                  {
                    "x": "504",
                    "y": "836"
                  },
                  {
                    "x": "826",
                    "y": "836"
                  },
                  {
                    "x": "826",
                    "y": "760"
                  }
                ]
              },
              {
                "languageCode": "en",
                "confidence": 0.9520227313
              },
              "text": "PENGUINS",
              "confidence": 0.9520227313
            }],
            "confidence": 0.9520227313
          },
          ...
        ]
      },
      "width": "1920",
      "height": "1280"
    }],
    ...
  ]
}
```

## Модерирование изображений

Классификация изображений сейчас находится на стадии Preview. Чтобы определить, соответствует ли изображение таким признакам, как контент для взрослых, шок-контент, водяные знаки, воспользуйтесь возможностью Классификации изображений. Для этого в методе `batchAnalyze` в свойстве `type` укажите `Classification`, а в конфигурации укажите модель `moderation`.

1) Подготовьте файл изображения, соответствующий требованиям:

- Поддерживаемые форматы файлов: JPEG, PNG, PDF.
- MIME-тип файла вы указываете в свойстве `mime_type`. По умолчанию `image`.
- Максимальный размер файла: 1 МБ.
- Размер изображения не должен превышать 20 мегапикселей (длина x ширина).

2) Кодировать файл в формат Base64:

3) Создайте файл с телом запроса, например `body.json`:

**body.json:**

```
{
  "folderId": "b1gvmob95yysaplct532",
  "analyze_specs": [{
    "content": "iVBORw0KGgo...",
    "features": [{
      "type": "CLASSIFICATION",
      "classificationConfig": {
        "model": "moderation"
      }
    }]
  }]
}
```

Где `analyze_specs: content` – изображение, кодированное в Base64.

4) Отправьте запрос с помощью метода `batchAnalyze` и сохраните ответ в файл, например `output.json`:

```
export IAM_TOKEN=CggaATEVAgA...
curl -X POST \
  -H "Content-Type: application/json" \
  -H "Authorization: Bearer ${IAM_TOKEN}" \
  -d '@body.json' \
  https://vision.api.cloud.yandex.net/vision/v1/batchAnalyze > output.json
```

В ответе будут содержаться признаки и вероятность соответствия этим признакам. По этим признакам вы можете модерировать изображение:

```
{
  "results": [
    {
      "results": [
        {
          "classification": {
            "properties": [
              {
                "name": "adult",
                "probability": 0.0017486262368038297
              },
              {
                "name": "gruesome",
                "probability": 0.0002884353743866086
              },
              {
                "name": "text",
                "probability": 0.13789896667003632
              },
              {
                "name": "watermarks",
                "probability": 0.99908816814422607
              }
            ]
          }
        }
      ]
    }
  ]
}
```

## Обнаружение лиц

Обнаружение лиц сейчас находится на стадии Preview. Чтобы обнаружить лица на фото, воспользуйтесь возможностью Обнаружение лиц. Для этого в методе batchAnalyze в свойстве type укажите FACE\_DETECTION.

1) Подготовьте файл изображения, соответствующий требованиям:

- Поддерживаемые форматы файлов: JPEG, PNG, PDF. MIME-тип файла вы указываете в свойстве mime\_type. По умолчанию image.
- Максимальный размер файла: 1 МБ.
- Размер изображения не должен превышать 20 мегапикселей (длина x ширина).

2) Кодировать файл в формат Base64:

3) Создайте файл с телом запроса, например body.json:

```
body.json:
{
  "folderId": "blgvmob95yysaplct532",
  "analyze_specs": [{
    "content": "iVBORw0KGgo...",
    "features": [{
      "type": "FACE_DETECTION"
    }]
  }]
}
```

Где analyze\_specs: content – изображение, кодированное в Base64.

4) Отправьте запрос с помощью метода batchAnalyze и сохраните ответ в файл, например output.json:

```
export IAM_TOKEN=CggaATEVAgA...
curl -X POST \
  -H "Content-Type: application/json" \
  -H "Authorization: Bearer ${IAM_TOKEN}" \
  -d '@body.json' \
  https://vision.api.cloud.yandex.net/vision/v1/batchAnalyze > output.json
```

### Готовая функция для отправки запросов в bash

Если у вас еще нет интерфейса командной строки Yandex Cloud, установите и инициализируйте его. Скопируйте в терминал функцию:

```
vision_face_detection() {
  curl -H "Authorization: Bearer `yc iam create-token`" \
    "https://vision.api.cloud.yandex.net/vision/v1/batchAnalyze" \
    -d @<(cat << EOF
{
  "folderId": "`yc config get folder-id`",
  "analyze_specs": [{
    "content": "`base64 -i $1`",
    "features": [{
      "type": "FACE_DETECTION"
    }]
  }]
}
EOF
)
}
```

Пояснения:

- yc iam create-token — получить IAM-токен.
- -d @<(cat << EOF ... EOF) — сформировать тело запроса.
- yc config get folder-id — получить ID каталога, выбранного по умолчанию в CLI.
- base64 -i \$1 — кодировать в Base64 изображение, переданное в аргументах функции.

Теперь вы можете вызывать эту функцию, передав путь к изображению в аргументах:

`vision_face_detection path/to/image.jpg`

## Задание 6. Yandex DataLens

Yandex DataLens — это сервис для бизнес-аналитики. Сервис позволяет подключаться к различным источникам данных, строить визуализации, собирать дашборды и делиться полученными результатами. С помощью Yandex DataLens вы можете отслеживать продуктовые и бизнес-метрики напрямую из источников, чтобы принимать решения, основанные на данных. Yandex DataLens не тарифицируется.

### Yandex DataLens

Сервис визуализации и анализа данных от [Yandex Cloud](#).

Сервис предоставляется бесплатно.

Без ограничений на количество пользователей и запросов.

[Открыть DataLens](#)

[Документация](#)



#### Визуализация данных и бизнес-аналитика

В несколько кликов создайте график, чтобы быстро проверить гипотезу на ваших данных. Соберите полноценный дашборд для мониторинга ключевых бизнес-метрик.

[Быстрый старт >](#)

#### Надёжно и безопасно

Отказоустойчивый, распределённый по трём дата-центрам сервис. Включён в реестр отечественного ПО. Соответствует требованиям ФЗ-152 и промышленным стандартам.

[Безопасность >](#)

#### Для всей команды

Добавьте учётные записи вашей команды или даже внешних партнёров для совместной работы. Благодаря тонкому разграничению прав доступа каждый увидит только свою аналитику.

[Подробнее >](#)

#### Marketplace расширений

Используйте дополнительные типы подключений, шаблоны дашбордов, датасеты и геослои для обогащения собственной аналитики.

[Marketplace >](#)

#### Работа с разными источниками данных

Подключайтесь к своим облачным и локальным базам данных, сервисам и плоским файлам. Комбинируйте данные из разных источников в одном дашборде.

[Подробнее >](#)

#### Геоаналитика на Яндекс.Картах

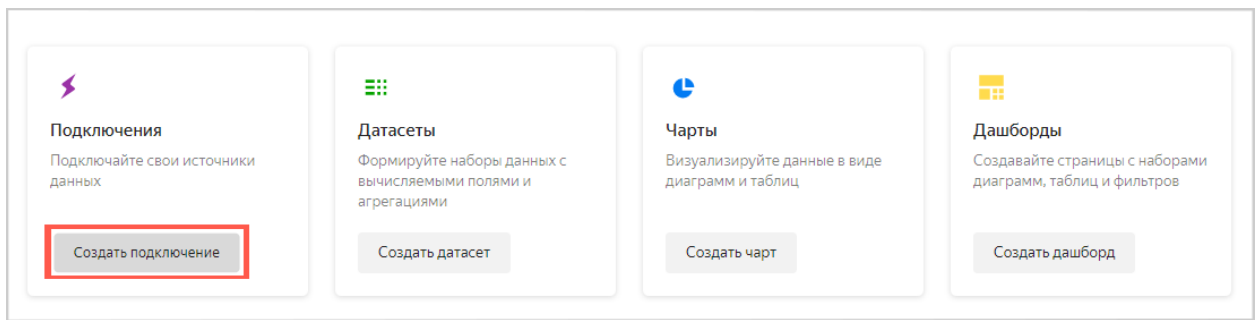
Используйте возможности Яндекс.Карт для вашей корпоративной аналитики. Подключите свой ключ API Яндекс карт для продвинутых возможностей геокодирования.

[Примеры >](#)

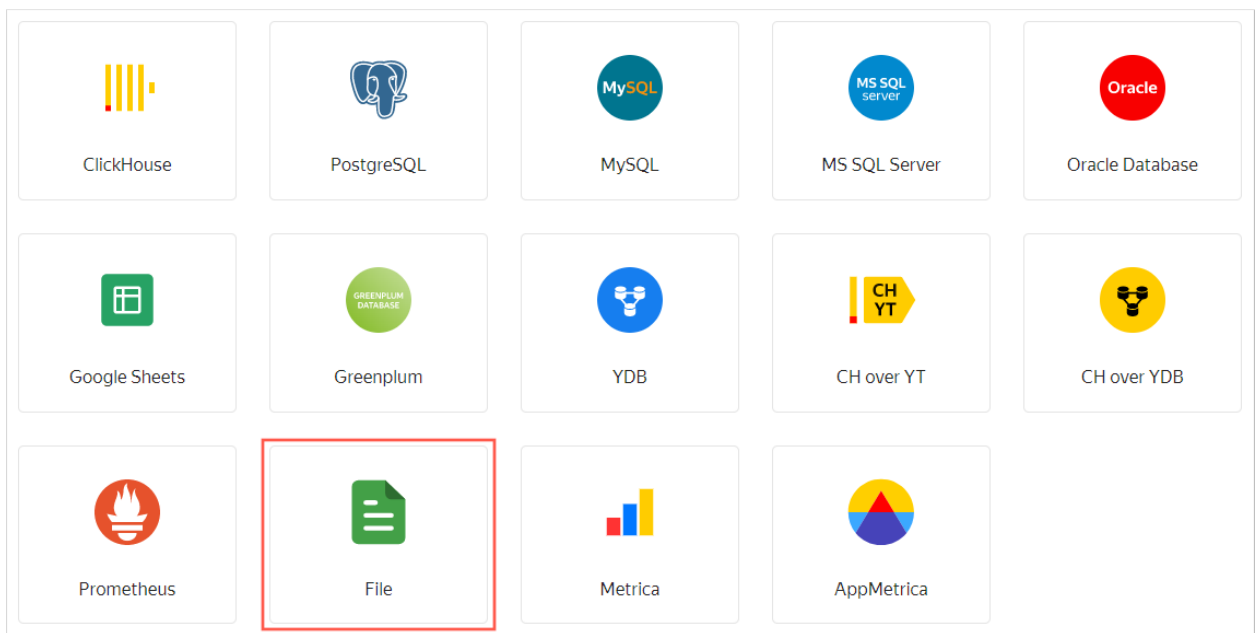
## Создайте подключение и датасет

Создайте датасет на основе подключения типа **File**.

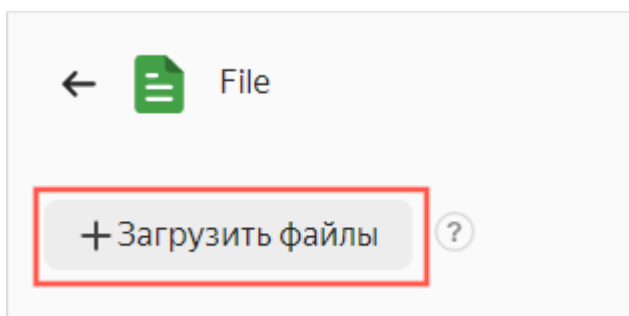
- 1) Перейдите на главную страницу DataLens.
- 2) Нажмите кнопку **Создать подключение**.



3) Выберите **File**.



4) Нажмите кнопку **Загрузить файлы**.



5) Выберите необходимый файл. В данном примере **SuperHeroes.csv** (<https://storage.yandexcloud.net/datalens/SuperHeroes.csv>). Дождитесь, когда на экране появится одержимое таблицы.

6) Нажмите кнопку **Создать подключение**.



Users / ... / Новое подключение Создать подключение

← File

SuperHeroes.csv ...

+ Загрузить файлы ?

Кодировка utf-8-sig
Разделитель Точка с запятой
Заголовок столбцов Да Нет

Name	Gender	Eye color	Race	Hair color	Height	Publisher	Skin color
Alien	Male	unknown	Xenomorph XX121	No Hair	244	Dark Horse Comics	black
Killer Frost	Female	blue	Human	Blond		DC Comics	blue
Mystique	Female	yellow (without irises)	Mutant	Red / Orange	178	Marvel Comics	blue
Nebula	Female	blue	Luphormoid	No Hair	185	Marvel Comics	blue
Abe Sapien	Male	blue	Ichthyo Sapien	No Hair	191	Dark Horse Comics	blue
Dr Manhattan	Male	white	Human / Cosmic	No Hair		DC Comics	blue
Shadow Lass	Female	black	Talokite	Black	173	DC Comics	blue
Beast	Male	blue	Mutant	Blue	180	Marvel Comics	blue
Archangel	Male	blue	Mutant	Blond	183	Marvel Comics	blue
Copycat	Female	red	Mutant	White	183	Marvel Comics	blue
Lobo	Male	red	Czarnian	Black	229	DC Comics	bluewhite

7) Введите имя подключения SuperHeroes и нажмите **Создать**.

## Создать подключение

×

>

SuperHeroes ×

Отменить
Создать

8) После сохранения нажмите кнопку **Создать датасет**.

### *Создайте датасет*

1) Если в рабочей области нет таблицы **SuperHeroes.csv**, то перетащите таблицу из панели выбора в рабочую область.

Users / / Новый датасет Создать чарт Сохранить

Источники Поля Параметры Фильтрация Предпросмотр

**Подключения**

SuperHeroes ...

**Таблицы**

SuperHeroes.csv

**Предпросмотр** Количество строк: 10 не более 10 000

#	Name	Gender	Eye color	Race	Hair color	Height	Publisher	Skin color	Alignment	Weight
1	Alien	Male	unknown	Xenomorph XX121	No Hair	244.0	Dark Horse Comics	black	bad	169.0
2	Killer Frost	Female	blue	Human	Blond		DC Comics	blue	bad	
3	Mystique	Female	yellow (without irises)	Mutant	Red / Orange	178.0	Marvel Comics	blue	bad	54.0

2) Перейдите на вкладку **Поля**.

Источники **Поля** Фильтрация

3) Создайте поле с показателем среднего веса супергероя:

- Нажмите значок в строке **Weight**.
- Выберите **Дублировать**.

8	Skin color	Skin color	Строка	Нет	...
9	Alignment	Alignment	Строка	Нет	...
10	<u>Weight</u>	Weight	Дробное число	Нет	...

**Предпросмотр** Количество строк: 10 не более 10 000

#	Name	Gender	Eye color	Race	Hair color	Height	Publisher	Skin color	Align
1	Alien	Male	unknown	Xenomorph XX121	No Hair	244.0	Dark Horse Comics	black	bad
2	Killer Frost	Female	blue	Human	Blond		DC Comics	blue	bad
3	Mystique	Female	yellow (without irises)	Mutant	Red / Orange	178.0	Marvel Comics	blue	bad

Дублировать  
Редактировать  
Удалить  
Права доступа

- Переименуйте дубликат поля **Weight (1)** в **Weight avg**: нажмите на имя строки, удалите текущее имя и введите новое.
- В столбце **Агрегация** для поля **Weight avg** выберите **Среднее**.

9	Alignment	Alignment	Строка	Нет	...
10	Weight	Weight	Дробное число	Нет	...
11	<u>Weight avg</u>	Weight	Дробное число	Нет	...

Предпросмотр    Количество строк: 10    не более 10 000

#	Name	Gender	Eye color	Race	Hair color	Height	Publ	Alignment	Weight
1	Alien	Male	unknown	Xenomorph XX121	No Hair	244.0	Dark	bad	169.0
2	Killer Frost	Female	blue	Human	Blond		DC C	bad	
3	Mystique	Female	yellow (without irises)	Mutant	Red / Orange	178.0	Marv	bad	54.0

Нет

Количество

Количество уникальных

Максимум

Минимум

**Среднее**

Сумма

4) В правом верхнем углу нажмите кнопку **Сохранить**.

default	cloud-tpu-vm2012	Users / tpu-vm2012 / Новый датасет	Создать чарт	<b>Сохранить</b>
---------	------------------	------------------------------------	--------------	------------------

Источники    Поля    Фильтрация    Обновить поля    Предпросмотр    + Добавить поле    Имя поля

#	Имя	Источник поля	Тип	Агрегация	Описание	Отображать скрытые поля
1	Name	Name	Строка	Нет		<input checked="" type="checkbox"/>
2	Gender	Gender	Строка	Нет		<input checked="" type="checkbox"/>

5) Введите имя датасета **SuperHeroes dataset**, нажмите **Создать**.

6) После сохранения датасета нажмите **Создать чарт**.

default	cloud-tpu-vm2012	Users / tpu-vm2012 / SuperHeroes dataset	<b>Создать чарт</b>	Сохранить
---------	------------------	--	---------------------	-----------

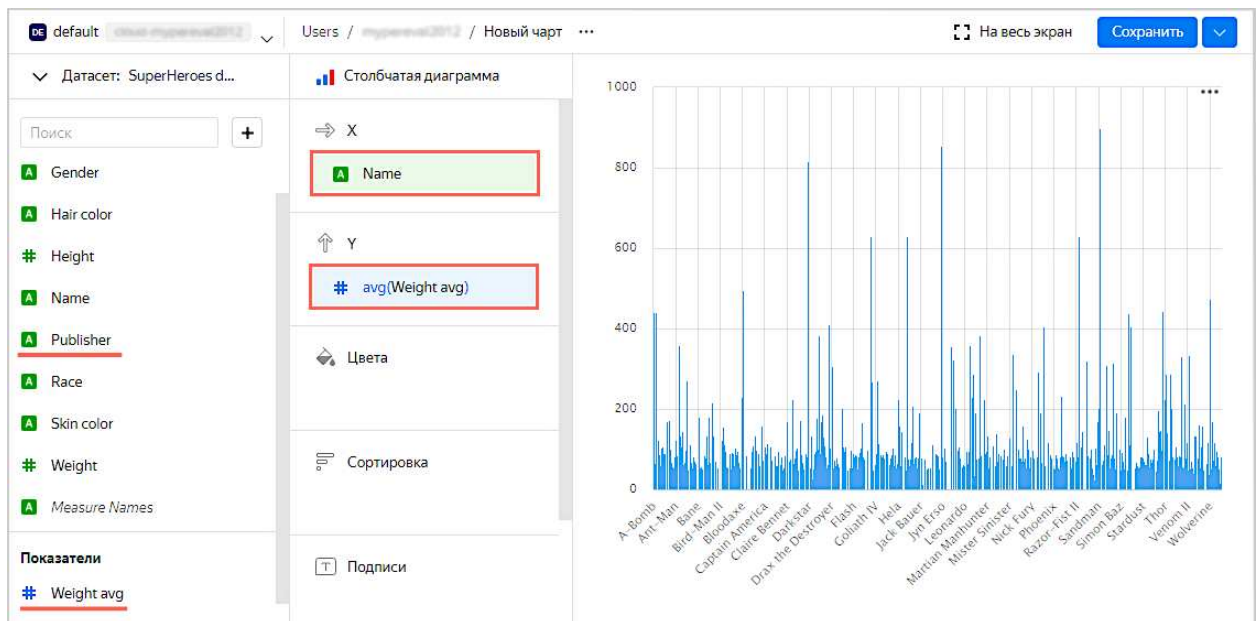
Источники    Поля    Фильтрация    Обновить поля    Предпросмотр    + Добавить поле    Имя поля

#	Имя	Источник поля	Тип	Агрегация	Описание	Отображать скрытые поля
1	Name	Name	Строка	Нет		<input checked="" type="checkbox"/>
2	Gender	Gender	Строка	Нет		<input checked="" type="checkbox"/>
3	Eye color	Eye color	Строка	Нет		<input checked="" type="checkbox"/>

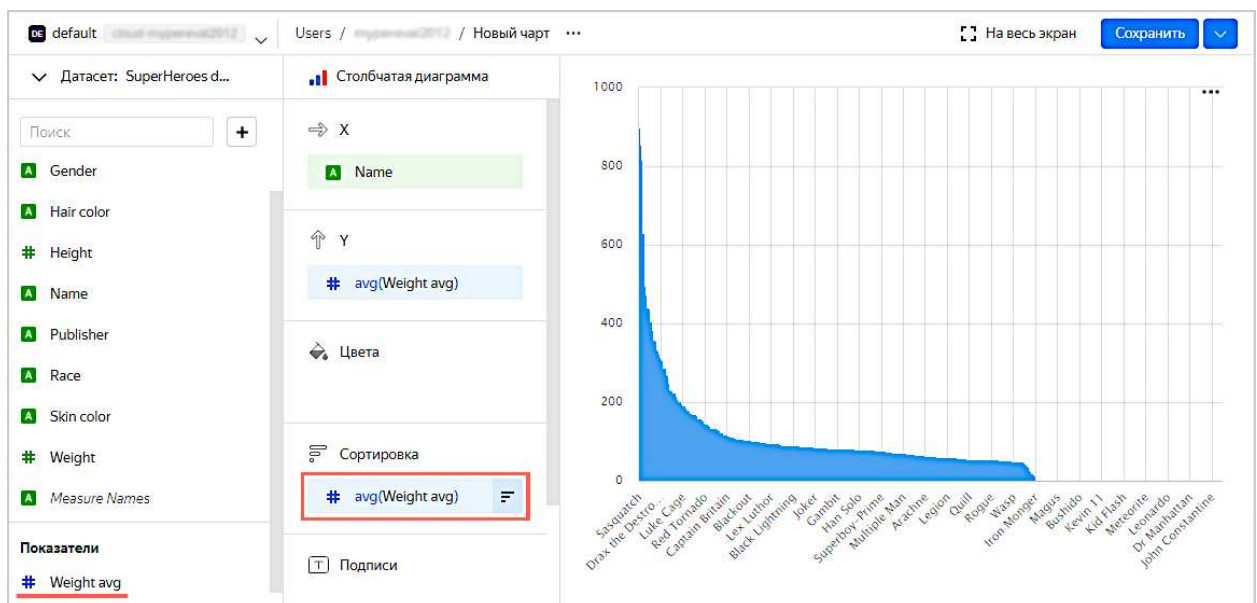
## Создайте чарт

Для визуализации разделения по половому признаку создайте чарт — столбчатую диаграмму.

- Добавьте на график имена супергероев. Для этого из раздела **Измерения** перетащите поле **Name** в секцию **X**.
- Добавьте на график веса супергероев. Для этого из раздела **Показатели** перетащите поле **Weight avg** в секцию **Y**.



Отсортируйте график по весам, например, по убыванию. Из первой колонки из раздела **Показатели** перетащите поле **Weight avg** в секцию **Сортировка**.



## Задание 7. Cloud functions

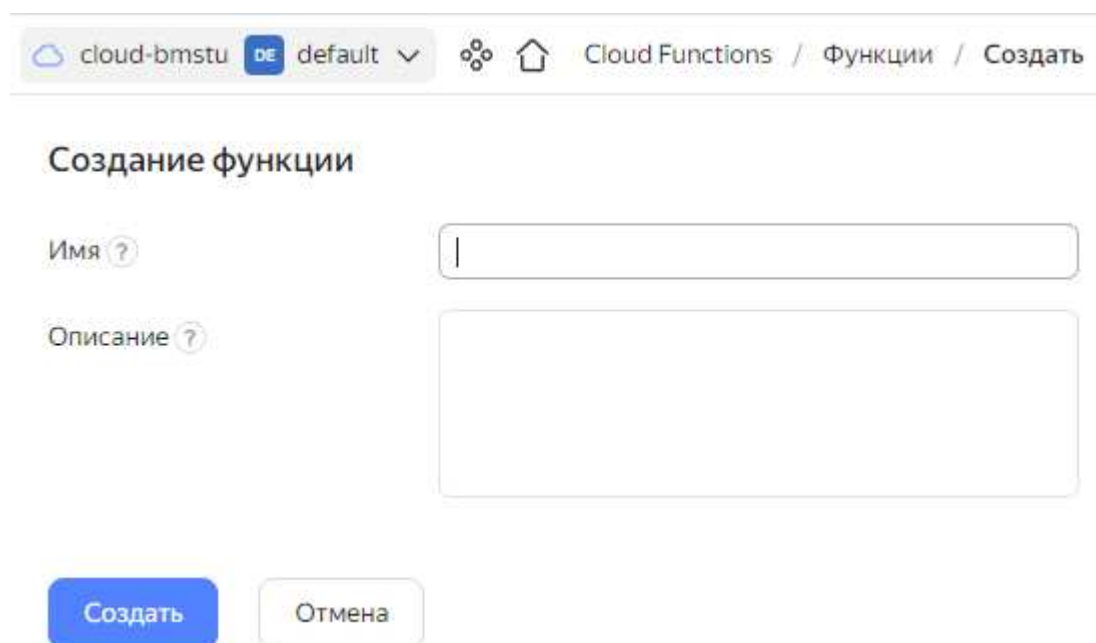
**Yandex Cloud Functions** (Functions as a Service, FaaS) позволяют нам заhostить код (Python, PHP, Net, NodeJS, R) на серверах (облачный хостинг, Platform as a Service, PaaS) Яндекса не разбираясь во всех сложностях DevOps.

- Не нужно покупать сервер, настраивать, обслуживать.

- Не нужно покупать домен и SSL сертификаты для того чтобы получить доступ к коду по HTTP(s).
- Не нужно разворачивать фреймворки вроде Django или Laravel.
- Не нужно платить за сервер, когда он не используется.

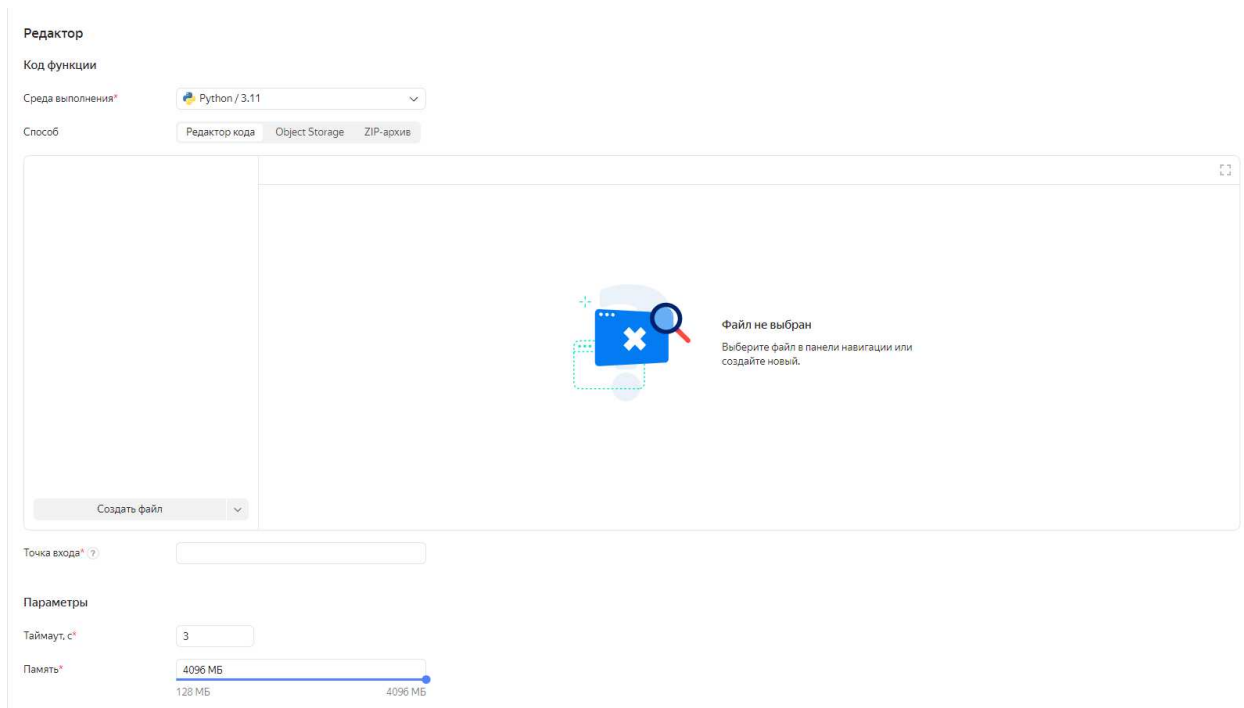
В случае с Cloud Functions оплата производится только за вызовы функции и время её работы. Конкретно в Yandex Cloud использование функций может быть вообще бесплатным при вполне приличном объеме использования.

А порой очень важно что вам не нужно ничего делать даже если нагрузка на проект построенный на Yandex Cloud Functions вдруг резко (возможно кратковременно вырастет) — все само по себе масштабируется.



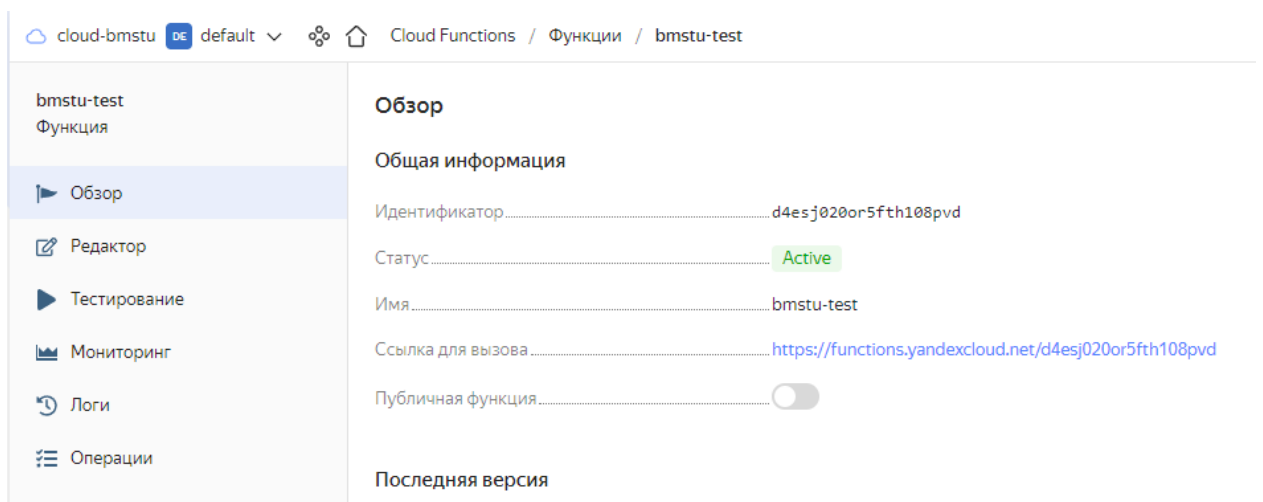
The screenshot shows the 'Создание функции' (Create Function) form in the Yandex Cloud console. The breadcrumb navigation at the top reads 'Cloud Functions / Функции / Создать'. The form includes a header 'Создание функции', a label 'Имя ?' (Name) with a text input field, and a label 'Описание ?' (Description) with a larger text area. At the bottom, there are two buttons: 'Создать' (Create) in blue and 'Отмена' (Cancel) in light gray.

Далее откроется окно с редактором исходного кода

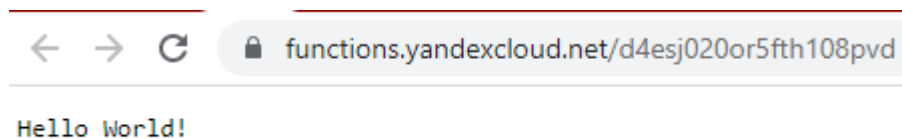


Исходный код также может быть загружен через object storage и zip архив. В точку входа можно прописать любую функцию из любого файла (<имя файла>.<название функции>), но она должна принимать на вход определенные параметры – (event, context)

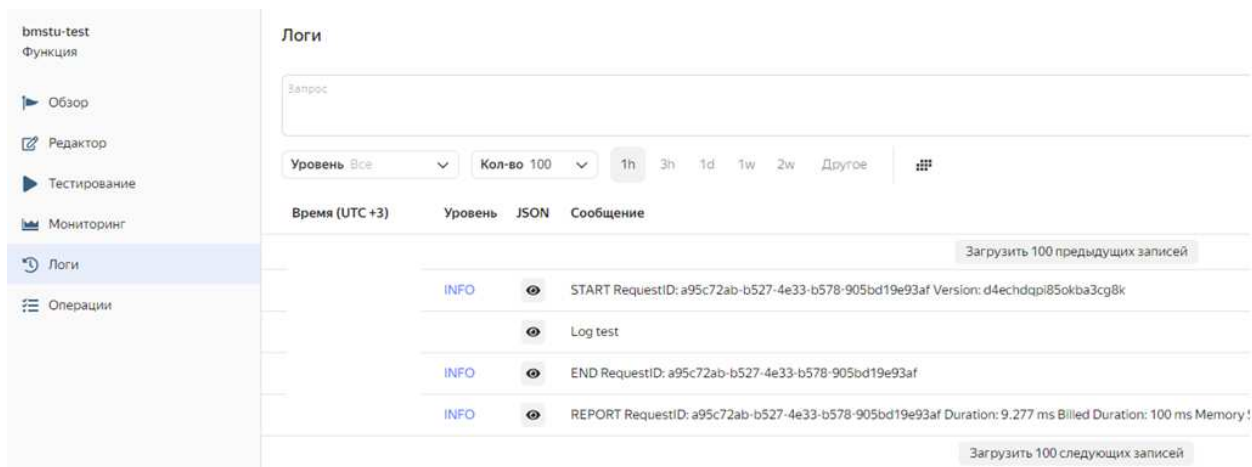
Чтобы функция была доступна по ссылке, ее надо сделать публичной:



Результат работы программы будет доступен по ссылке для вызова



При необходимости просмотра вывода print результат работы функции можно посмотреть во вкладке логи.



Для установки зависимостей необходимо создать файл requirements.txt и прописать в нем необходимые модули. Более подробно с форматом файла requirements.txt можно ознакомиться в документации pip.

## ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ

Решите задачу согласно варианту с использованием платформы Yandex Cloud.

### Варианты

1. На основе сервиса cloud functions создать сервис предсказания погоды на небольшой промежуток времени (каждый день в неделе или каждый час дня), прогноз выдается при обращении к странице по адресу (при выборе реализации предсказания в первую очередь ориентироваться на ограничения Yandex Cloud, а не на показатели точности).
2. С помощью синтеза речи Yandex Speech озвучить машинно-сгенерированный текст (использовать APIv3)
3. С помощью синтеза речи Yandex Speech озвучить машинно-сгенерированный текст (использовать APIv1)
4. С помощью сервиса Yandex Cloud Translate перевести на русский язык машинно-сгенерированный текст на английском языке
5. С помощью сервиса Yandex DataLens подготовить визуализацию распределения числовых параметров в выбранном датасете
6. С помощью сервиса Yandex DataLens подготовить визуализацию распределения числовых параметров в одном из тестовых наборов ClickHouse
7. С помощью распознавания текста на изображении сервиса Yandex Vision создать приложение с графическим интерфейсом с возможностью загрузки изображения и отображения имеющегося на нем текста, в случае отсутствия текста на изображении сообщить пользователю об этом.
8. С помощью распознавания текста на pdf сервиса Yandex Vision приложение с графическим интерфейсом с возможностью загрузки файла и отображения имеющегося в нем текста, в случае отсутствия текста в файле сообщить пользователю об этом.



9. С помощью распознавания лиц на изображении сервиса Yandex Vision создать приложение верификации изображения для пропуска программа должна иметь графический интерфейс с возможностью загрузки изображения, в ответ на которое будет выводиться, что загруженное изображение подходит, если на нем распознано 1 лицо, иначе изображение не подходит).
10. С помощью модерирования изображения сервиса Yandex Vision создать приложение проверки загружаемой аватарки (программа должна иметь графический интерфейс с возможностью загрузки изображения, в ответ на которое в зависимости от ответа сервиса yandex vision будет выдвигаться решение о допуске или недопуске к использованию данного изображения)

### **ТРЕБОВАНИЯ К РЕАЛИЗАЦИИ**

Для реализации элементов машинного обучения в части задания, не связанного с сервисами Yandex Cloud допускается использование сторонних классов и компонентов. По завершении готовится отчет.

### **КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ**

1. Какие сервисы входят в состав Yandex Cloud?
2. Что такое метка в рамках платформы Yandex Cloud?
3. Что представляет собой Виртуальная машина в рамках платформы Yandex Cloud?
4. В каких статусах может находиться виртуальная машина?
5. Какие типы ресурсов можно подключать к виртуальным машинам?
6. Опишите возможности ClickHouse.
7. Перечислите шаги по созданию аккаунта Yandex Cloud.
8. Каким образом можно получить IAM токен?

## **ФОРМА ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ**

На выполнение лабораторной работы отводится 2 занятия (4 академических часа: 3 часа на выполнение и сдачу лабораторной работы и 1 час на подготовку отчета).

Номер варианта студенту выдается преподавателем.

Отчет на защиту предоставляется в печатном виде.

Структура отчета (на отдельном листе(-ах)): титульный лист, формулировка задания (вариант), листинг, результаты выполнения работы (графические изображения примеров работы приложения), выводы).

## ОСНОВНАЯ ЛИТЕРАТУРА

1. Зиангирова, Л. Ф. Технологии облачных вычислений [Электронный ресурс]: учебное пособие / Л. Ф. Зиангирова. — Электрон, текстовые данные.— Саратов : Вузовское образование, 2016. — 300 с. — Режим доступа <http://www.iprbookshop.ru/41948.html>

2. Клементьев, И. П. Введение в облачные вычисления [Электронный ресурс]/ И. П. Клементьев, В. А. Устинов. — Электрон, текстовые данные. — Москва: Интернет- Университет Информационных Технологий (ИНТУИТ), 2016. — 298 с. — Режим доступа: <http://www.iprbookshop.ru/57372.html>

3. Савельев, А. О. Введение в облачные решения Microsoft [Электронный ресурс]/ А. О. Савельев. — Электрон, текстовые данные.— 2-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 230 с. — Режим доступа: <http://www.iprbookshop.ru/73665.html>

4. Рак, И.П. Технологии облачных вычислений [Электронный ресурс]: учебное пособие / И.П. Рак, А.В. Платёнкин, Э.В. Сысоев ; Министерство образования и науки Российской Федерации, Федеральное государственное бюджетное образовательное учреждение высшего образования «Тамбовский государственный технический университет». - Тамбов Издательство ФГБОУ ВПО «ТГТУ», 2017.— 82 с. : ил. -URL: <http://biblioclub.ru/index.php?page=book&id=499410>

5. Тропченко, А.А. Методы вторичной обработки и распознавания изображений [Электронный ресурс]: учебное пособие / А.А. Тропченко, А.Ю., Тропченко. — СПб.: Университет ИТМО, 2015. — 215 с. — Режим доступа: <http://www.iprbookshop.ru/67277.html>

6. Шапиро, Л. Компьютерное зрение [Электронный ресурс]: учебное пособие / Л. Шапиро, Д. Стокман. — М.: Издательство "Лаборатория знаний", 2015. — 763 с. — Режим доступа: <https://e.lanbook.com/book/84096>

## ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

8. Березовская, Е. А. Работа с сервисом бизнес-аналитики Yandex DataLens: учебное пособие / Е. А. Березовская, С. В. Крюков. — Ростов-на-Дону: ЮФУ, 2022. — 92 с. — Режим доступа: <https://e.lanbook.com/book/297083>.

9. Гинько, А. Ю. Анализ и визуализация данных в Yandex DataLens. Подробное руководство: от новичка до эксперта: руководство / А. Ю. Гинько. — Москва: ДМК Пресс, 2022. — 356 с. — Режим доступа: <https://e.lanbook.com/book/314909>.

10. Дружинин, Д. В. Высокопроизводительные вычисления и облачные технологии: учебное пособие / Д. В. Дружинин. — Томск: ТГУ, 2020. — 94 с. — Режим доступа: <https://e.lanbook.com/book/202355>.

12. Мирзоев, М.С. Основы математической обработки информации [Электронный ресурс]: учебное пособие / М.С. Мирзоев. — М.: Прометей, 2016. — 316 с. — 978-5-906879-01-1. — Режим доступа: <http://www.iprbookshop.ru/58165.html>

11. Попок, Л. Е. Технологии облачных вычислений : учебное пособие / Л. Е. Попок, Д. А. Замотайлова, Д. Н. Савинская. — Краснодар: КубГАУ, 2019. — 66 с. — Режим доступа: <https://e.lanbook.com/book/254231>.