



Проект «Открытый университет»

НИУ ВШЭ – ПЕРМЬ
МИНИСТЕРСТВО
ОБРАЗОВАНИЯ И НАУКИ
ПЕРМСКОГО КРАЯ

Пермь 2023



Ланин Вячеслав Владимирович

академический руководитель программы «Разработка информационных систем для бизнеса»

Как работают чат-боты и кто их разрабатывает?

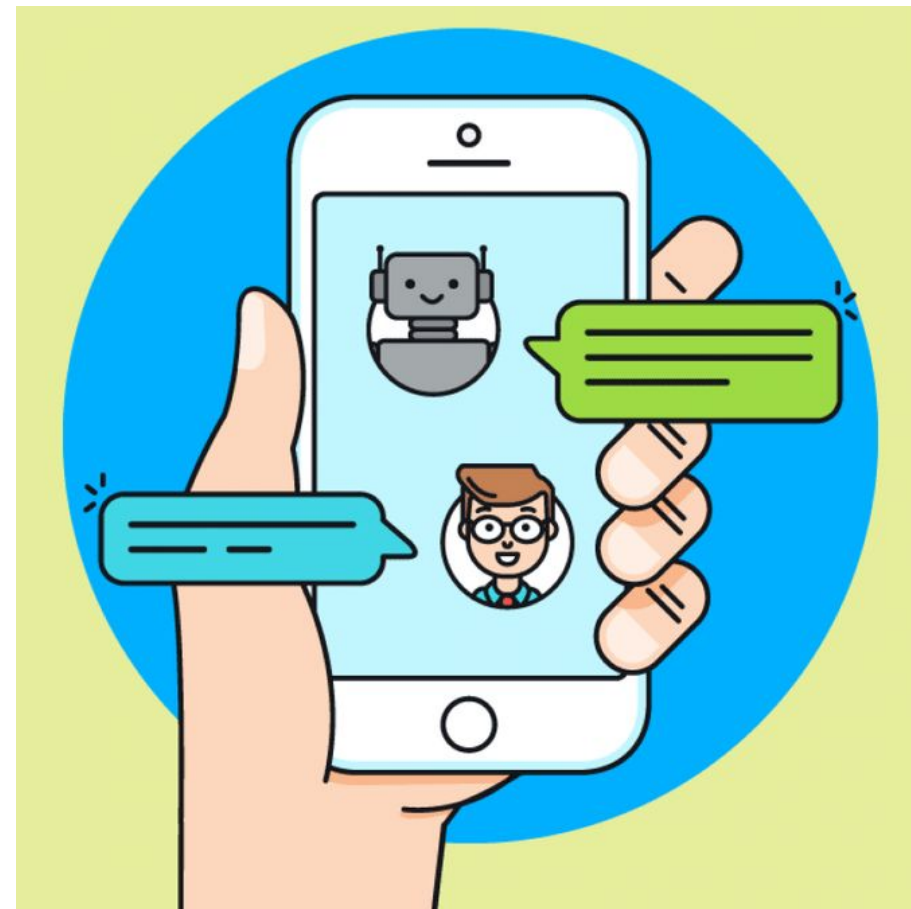
В Что такое чат-бот?

Диалоговые системы (разговорные интерфейсы, диалоговые агенты) – это компьютерные системы, с которыми пользователь взаимодействует на естественном языке (ЕЯ): письменном, устном или комбинируя оба способа.

Чат-бот (виртуальный собеседник или просто бот) — это компьютерная программа, которая взаимодействует с пользователями на естественном языке.

В основе работы таких лежат механизмы обработки естественного языка (NLP) и искусственный интеллект.

Чат-боты могут понимать человеческий язык, имитировать разговоры и выполнять автоматические задачи.

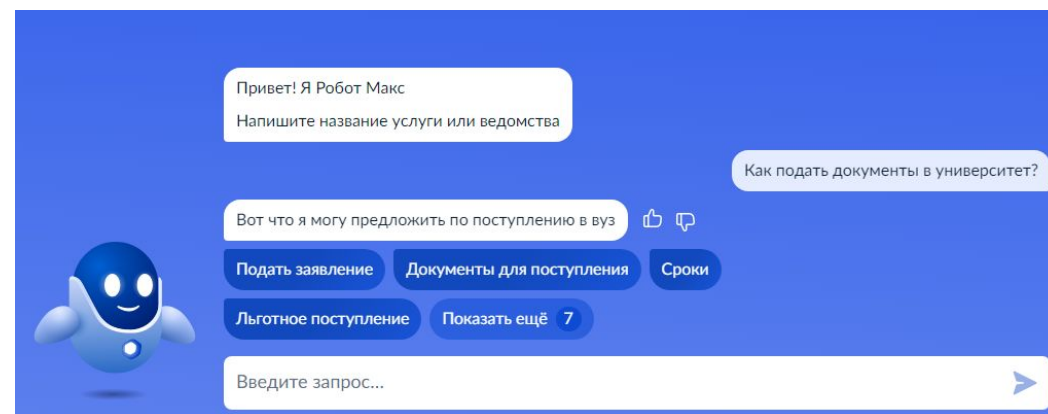


В Как работают чат-боты?

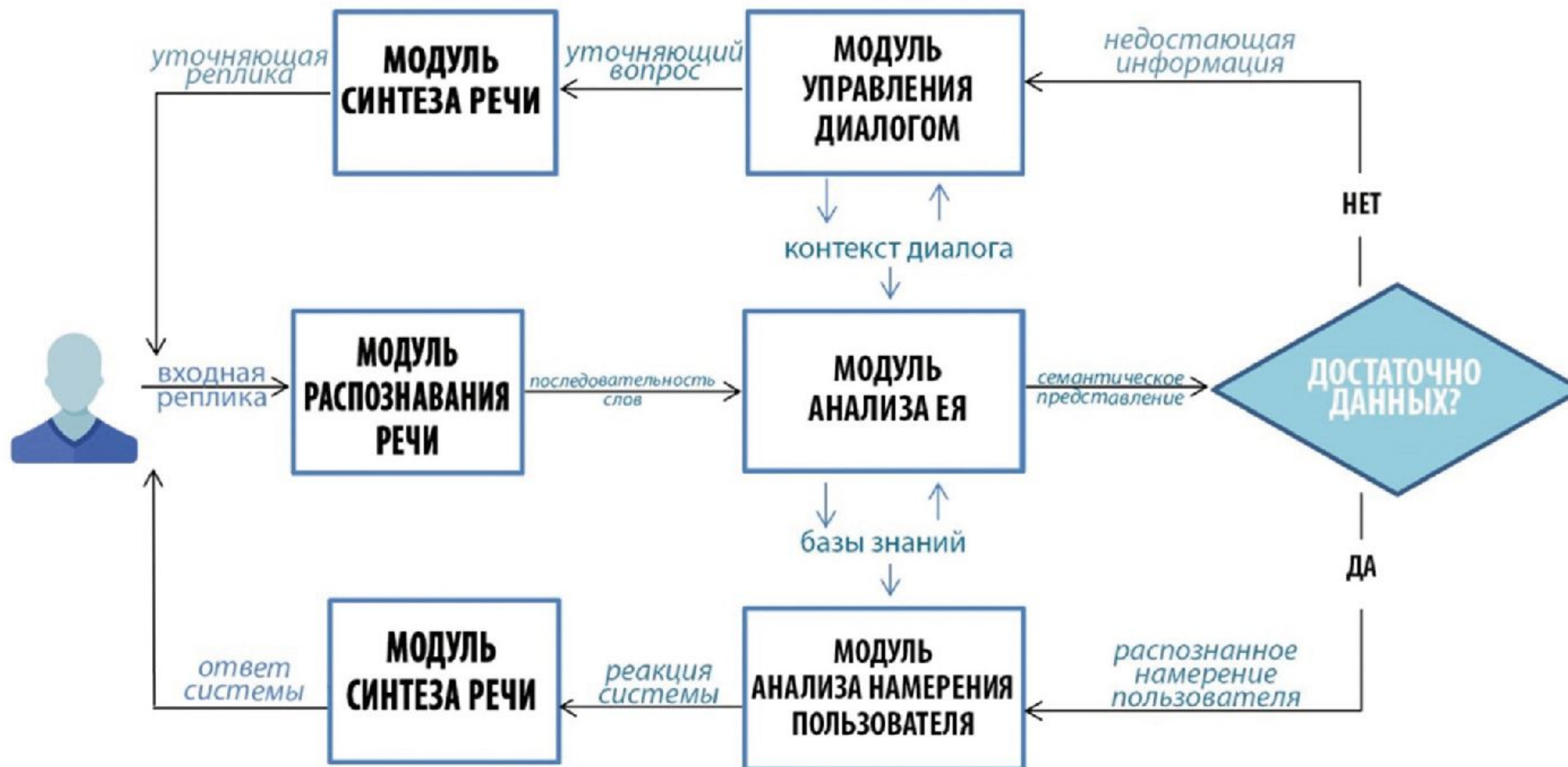
Шаг 1. Пользователь отправляет чат-боту текстовое или голосовое сообщение через любой канал, например приложение или веб-сайт. Сообщение может быть в формате команды или вопроса.

Шаг 2. Чат-бот получает сообщение и записывает связанную информацию, например канал взаимодействия, а затем с помощью алгоритмов NLP определяет цель сообщения и намерения.

Шаг 3. Чат-бот определяет соответствующий ответ и возвращает его пользователю через тот же канал. Остальной диалог с чат-ботом происходит по тому же принципу. Беседа продолжается до тех пор, пока пользователь не получит нужный ответ или пока обращение не будет переадресовано агенту.



Архитектура современной диалоговой системы





Задачи компьютерной лингвистики

- Машинный перевод
- Информационный поиск
- Классификация и кластеризация текстов
- Реферирования и аннотирование текстов
- Формирование ответов на вопросы
- Автоматизация подготовки и редактирования текстов
- Извлечение информации из текстов
- Генерация текстов на ЕЯ
- Организация диалога с пользователем на ЕЯ
- Обучение ЕЯ
- Распознавание и синтез звучащей речи

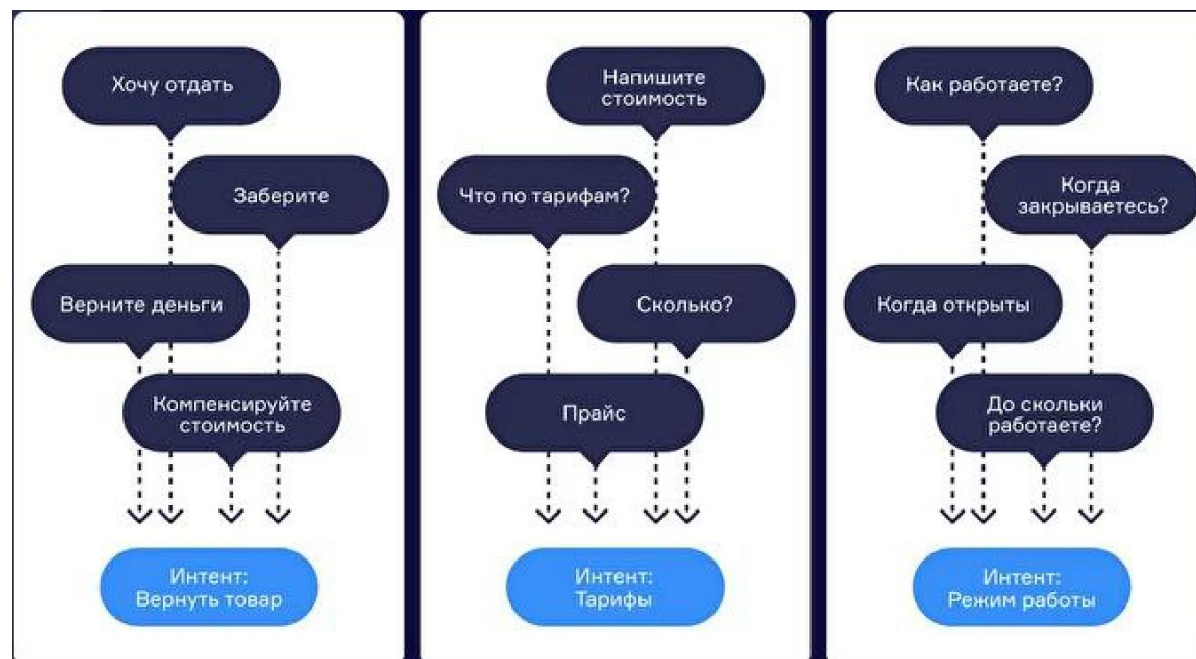
В Модели диалога

Выделяют два основных подхода к моделированию диалога.

Использование диалоговой грамматики. Взаимодействие агентов строится на предположении о том, что диалог как четко структурированная система: приветствие-приветствие, вопрос-ответ, предложение-принятие/отклонение.

Методы, основанные на интентах. Коммуникация определена целями участников диалога.
Интент

Интент — намерение или потребность пользователя.





Чем отличаются простые боты от «умных»?

Что говорит пользователь	Что понимает обычный бот	Что понимает умный бот с ИИ
Привет!	Пользователь здоровается ✓	Пользователь здоровается ✓
Даров!	Непонятно !	Пользователь здоровается ✓
Бонжур, дорогой!	Непонятно !	Пользователь здоровается ✓
Привет! Хочу вернуть товар	Пользователь здоровается и хочет вернуть товар ✓	Пользователь здоровается и хочет вернуть товар ✓
Привет! Товар бракованный, отдайте деньги!	Пользователь здоровается, дальше непонятно !	Пользователь здоровается и хочет вернуть товар ✓



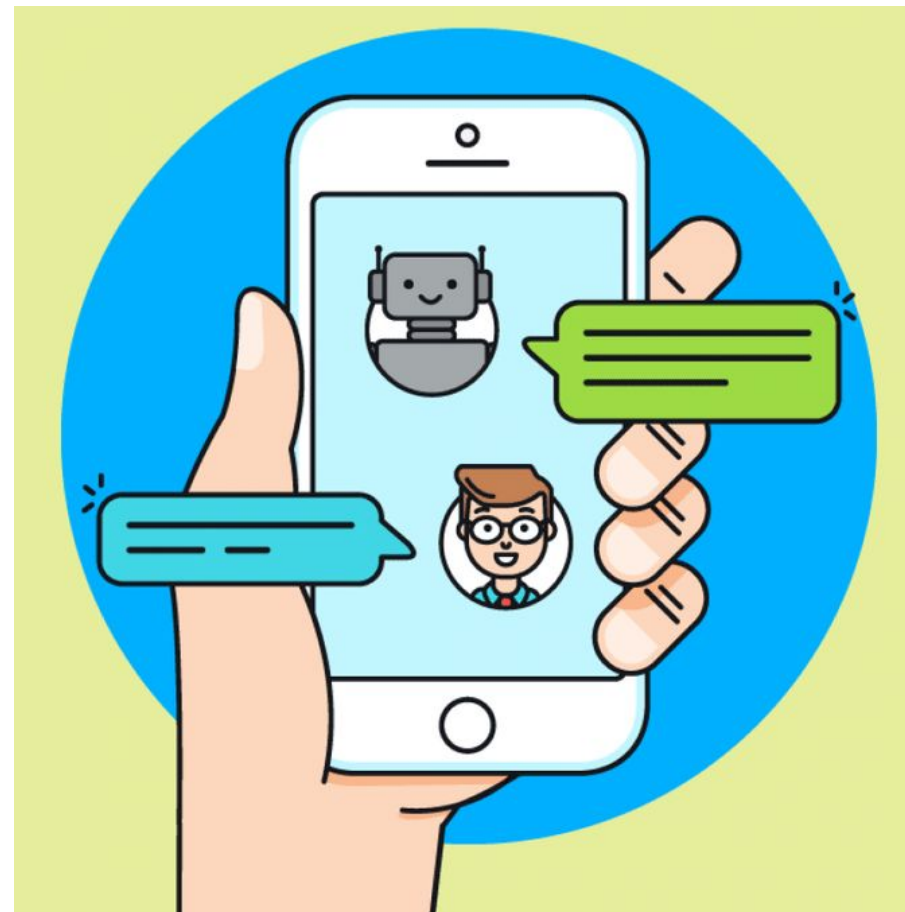
Какие способы создания ботов существуют?

Готовые боты и шаблоны — самый быстрый способ внедрить виртуального помощника в свой проект. Подходит для небольших и узконаправленных задач и не рассчитан на настройку.

Конструкторы ботов — способ создать бота в визуальном интерфейсе по собственному сценарию и адаптировать его под требования проекта без специальных навыков программирования.

Диалоговые платформы — способ, который позволяет создавать масштабные решения с продвинутой логикой. Требуется навыки программирования и дополнительных ресурсов.

Специализированные фреймворки — инструменты для разработчиков, позволяющий описывать логику сценариев, не ограничиваясь архитектурой диалоговых платформ и других решений.



Социальные медиа

- Понятие «социальная сеть» появилось в 1954 г., ввел его социолог **Джеймс Барнс**.
- **Социальная сеть** — это социальная структура, состоящая из группы узлов, которыми являются социальные объекты (люди или организации), и связей между ними (социальных взаимоотношений).
- Основные понятия: **актор** и **связь**.
- С появлением Интернета научная концепция Джеймса Барнса стала набирать популярность.
- Под **социальными медиа** понимается категория веб-сервисов, предоставляющих возможность публикации, обмена и обсуждения контента широким кругом пользователей.

Виды социальных медиа

- 1. Направленные на общение** - социальные сети, блоги, сайты отзывов и знакомств, а также приложения и сайты связанные с геокоммуникациями
- 2. Медиа для совместного редактирования информации** – Википедия и другие приложения, предоставляющие возможность совместного редактирования в приложения. Например, Яндекс.Карты (можно указать происшествия и события в городе и остальные пользователи будут видеть информацию на картах)
- 3. Мультимедийные социальные медиа** - порталы, на которых возможен обмен и комментирование мультимедийными файлами. Чаще всего это хостинги, на которых хранится информация с возможностью комментирования и редактирования.

В Почему чат-боты становятся популярными?

1. Пользователи «устали» загружать и устанавливать приложения (переутомление от приложений)
2. Люди стали чаще общаться при помощи текста, используя мессенджеры, люди массово переходят на службы сообщений и постоянно находятся в чатах.
3. Многие компании заинтересованы в анализе и автоматизации общения с клиентами,
4. Растет число «умных» бытовых предметов, которыми можно управлять.

«Интерес пользователей к установке новых приложений падает, они тратят все больше времени на использование тех приложений, которые у них уже есть»

(директор по исследованиям Gartner Джессика Экхольм)

«Мессенджеры самый удобный и безопасный способ коммуникации»

(продукт менеджер «Сбербанка» FutureBanking Юрий Герасименко)

В Какие существуют типы диалоговых систем?

Транзакционные чат-боты (целеориентированной системой, англ. goal/task-oriented) - это узкоспециализированные чат-боты, основная цель которых заключается в выполнении или автоматизации одной функции. Транзакционные чат-боты предназначены для ответа на фиксированные вопросы и решения определенных проблем. После выбора вопроса чат-бот будет предлагать возможные варианты, пока пользователь не получит нужный ответ или пока обращение не будет переадресовано агенту.

Разговорные чат-боты (чат-ориентированные, англ. chat-oriented) обладают более развитыми, интерактивными и персонализированным возможностями. Благодаря разговорному искусственному интеллекту, алгоритмам обработки естественного языка, доступу к базам знаний и другой информации эти чат-боты могут обнаруживать нюансы в вопросах и ответах пользователей, а затем давать точные динамические ответы так, как если бы они были живыми людьми.



Как чат-ботов используют в различных отраслях?

- Электронная коммерция и торговля
- Финансовые услуги
- здравоохранение
- Образование
- Страхование
- Производство
- Путешествия и гостиничный бизнес

Чат-бот ChatGPT

ChatGPT — это чат-бот с применением искусственного интеллекта, поддерживающий запросы на естественных языках.

Основные возможности:

- Поиск информации и ответы на вопросы.
- Генерация диалогов и рассказов.
- Генерация описательных текстов и рефератов.
- Генерация программного кода.



OpenAI

Лексико-семантические шаблоны

- **Лексико-синтаксический** (семантический) шаблон – декларативная структура, структурный образец языковой конструкции, который отображает её лексические и поверхностные синтаксические свойства



Регулярные выражения

Регулярные выражения (англ. *regular expressions, RegEx*) — технология сопоставления текстовых фрагментов шаблону, записанному на специальном языке.

Регулярные выражения используются для сжатого описания некоторого множества строк с помощью шаблонов, без необходимости перечисления всех элементов этого множества.

История регулярных выражений

- 50-е годы прошлого века в математике (теория автоматов).
- Использование в ОС Unix (grep).
- Язык Perl (поддержка встроена непосредственно в интерпретатор).
- В настоящее время регулярные выражения используются повсеместно: Java, JScript, C/C++, .NET.

Задачи, решаемые с помощью регулярных выражений

1. поиск и извлечение подстроки, удовлетворяющей шаблону регулярного выражения.
2. поиск и замена подстроки, удовлетворяющей шаблону регулярного выражения.
3. проверки на соответствие заданной строки шаблону.

язык регулярных выражений

Для записи регулярных выражений используется специальный синтаксис - **язык регулярных выражений**.

Язык описания регулярных выражений состоит из символов двух видов:

- **обычный символ** представляет в выражении сам себя;
- **метасимвол** представляет некоторый класс символов.

Простейшее регулярное выражение

Простейшее регулярное выражение состоит из **обычных СИМВОЛОВ**.

Пример:

кошка

Кошка кошечка **кошками** кошки **кошка** кошечка

Метасимволы

Метасимвол – символ, имеющий специальное значение в регулярном выражении.

Метасимвол может быть представлен одним или несколькими символами:

\wedge – начало строки

\cdot – (точка) произвольный символ

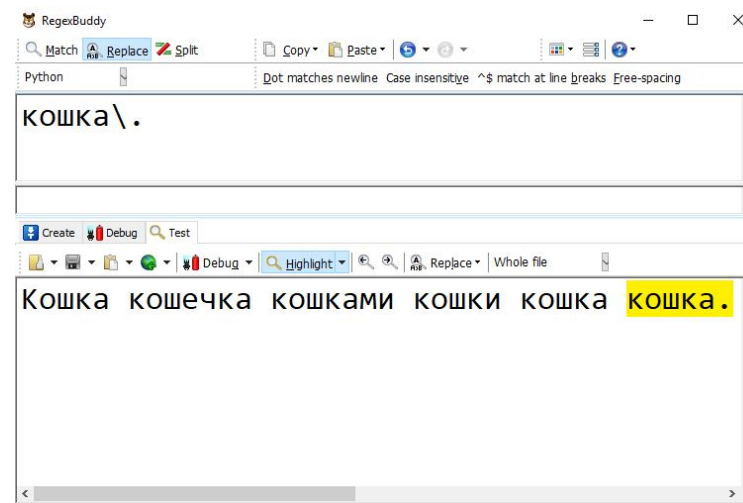
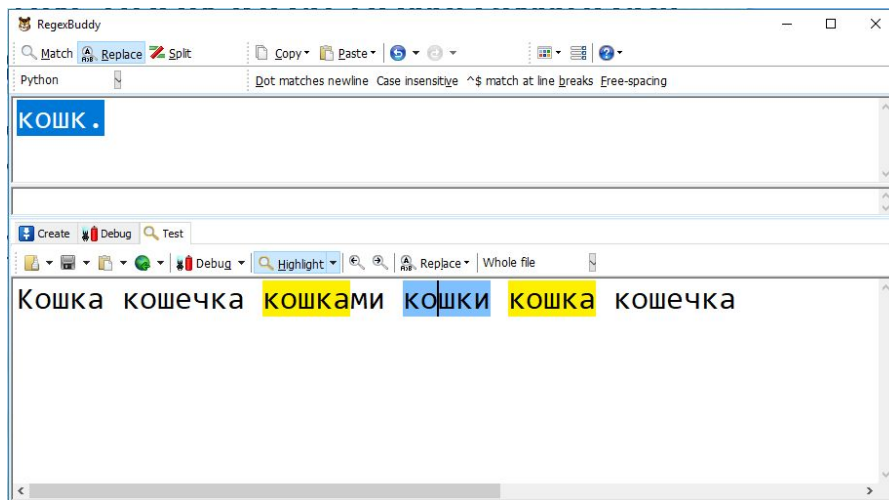
$\backslash d$ – символ из множества цифр

$\{n,m\}$ – задает, по меньшей мере, n , но не более m
соответствий

Соответствие произвольному символу: метасимвол «.» (точка)

Подстановочный знак: соответствует **любому одиночному символу**, кроме `\n` (перевод строки).

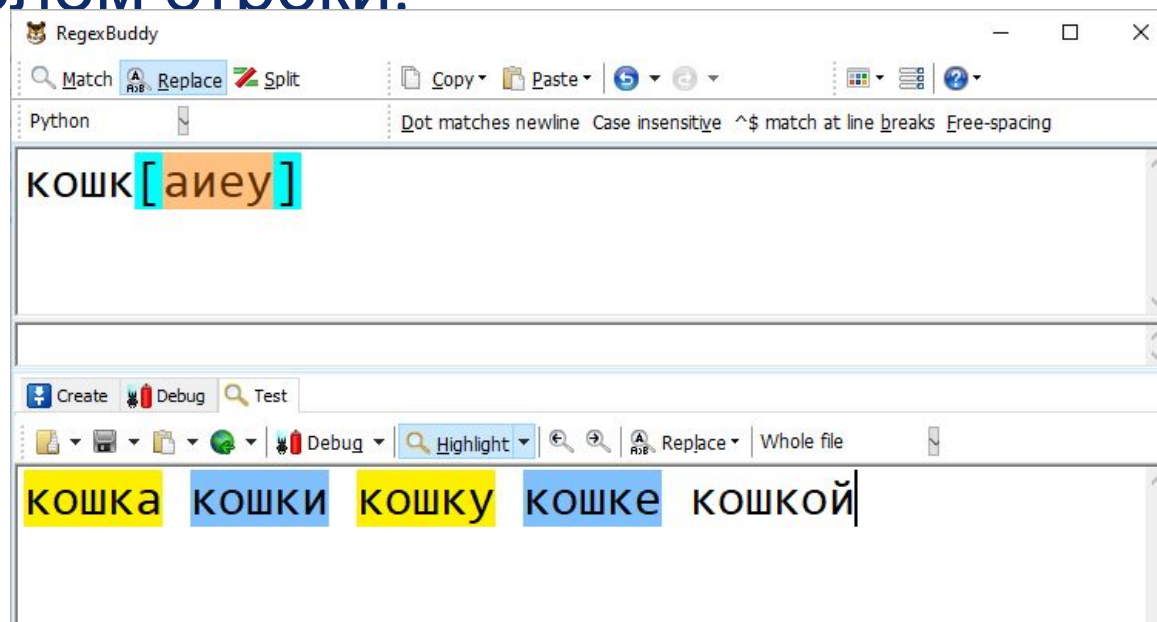
Для сопоставления символа точки («.» или `\u002E`) перед ней нужно поставить дополнительную обратную косую черту (`\.`).



Соответствие одному из нескольких СИМВОЛОВ

Набор символов определяется метасимволами [и].

Символ соответствует набору, если этот символ входит в набор. При сопоставлении шаблона со строкой весь набор сопоставляется с одним СИМВОЛОМ СТРОКИ.



Диапазон набора символов

Для упрощения работы с символьными диапазонами используется метасимвол - (дефис).

- [0-9] соответствует [0123456789]
- [5-7] соответствует [567]
- [A-Z] все заглавные буквы латинского алфавита
- [A-z] все символы латинского алфавита
- [3-1] недопустимо
- [^набор] отрицание символов в наборе

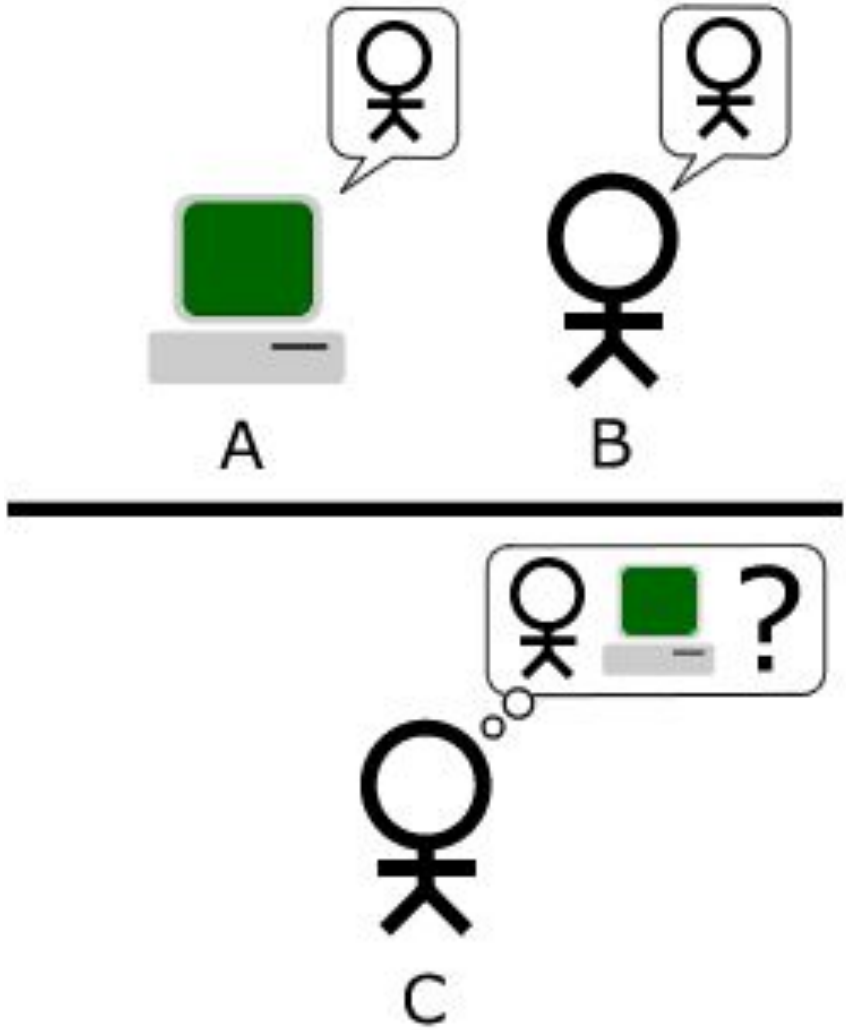
предопределенные последовательности

- `\d` Соответствует любой цифре; эквивалент класса `[0-9]`.
- `\D` Соответствует любому нечисловому символу; эквивалент класса `[^0-9]`.
- `\s` Соответствует любому символу `whitespace`; эквивалент `[\t\n\r\f\v]`.
- `\S` Соответствует любому не-`whitespace` символу; эквивалент `[^\t\n\r\f\v]`.
- `\w` Соответствует любой букве или цифре; эквивалент `[a-zA-Z0-9_]`.
- `\W` эквивалент `[^a-zA-Z0-9_]`

Символы повторения (Квантификаторы)

Метасимвол	Описание
{n,m}	Соответствует предшествующему шаблону, повторенному не менее n и не более m раз
{n,}	Соответствует предшествующему шаблону, повторенному n или более раз
{n}	Соответствует в точности n экземплярам предшествующего шаблона
?	Соответствует нулю или одному экземпляру предшествующего шаблона; предшествующий шаблон является необязательным
+	Соответствует одному или более экземплярам предшествующего шаблона
*	Соответствует нулю или более экземплярам предшествующего шаблона

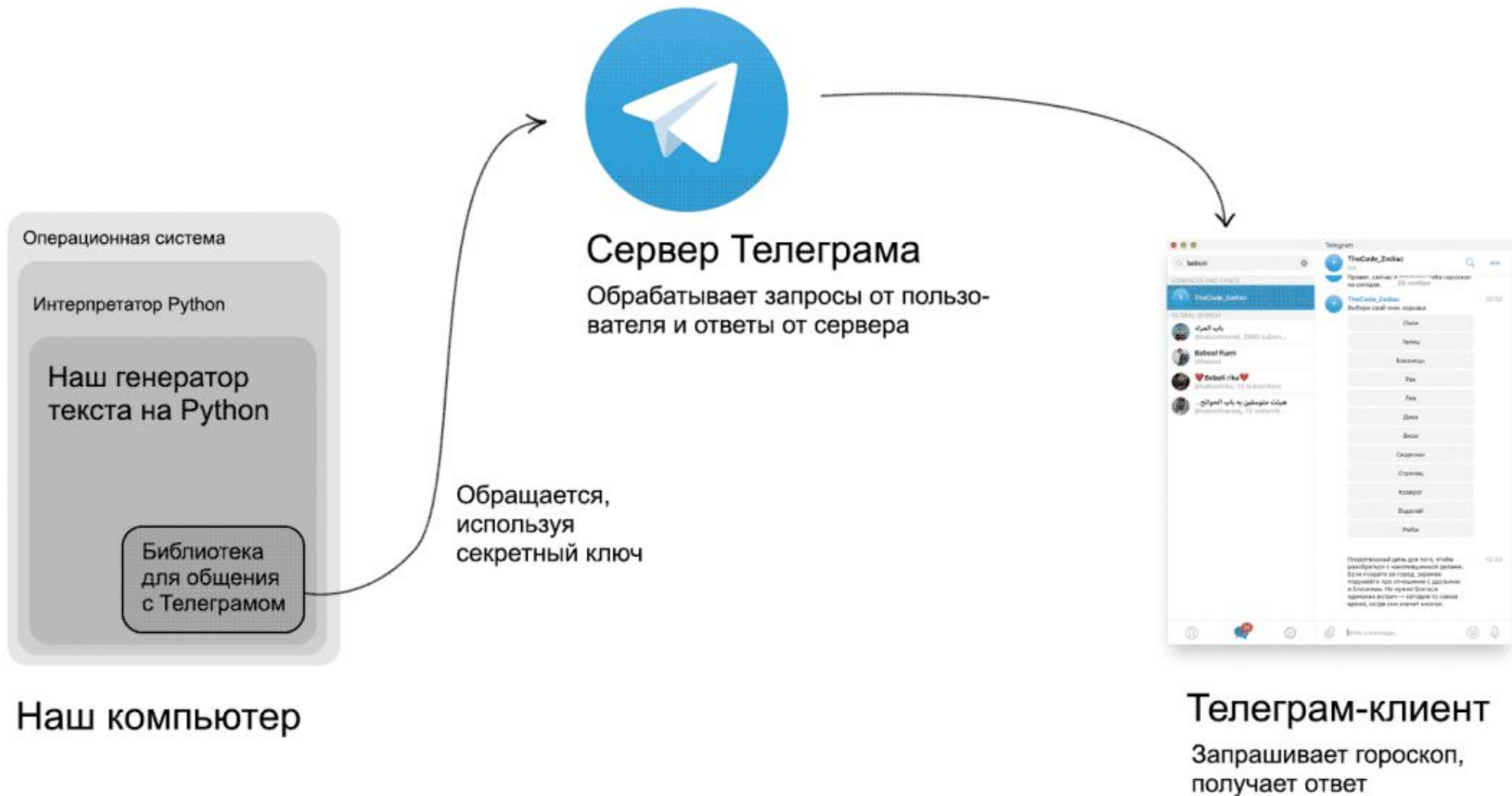
Тест Тьюринга



Регистрация бота

1. Найти бота @BotFather
2. Выполнить команду /start или /newbot
3. Указать название бота и логин.
4. Получить и сохранить токен бота.

Архитектура telegram-бота



API Telegram и специализированные библиотеки

1. Документация: <https://core.telegram.org/bots/api>
2. Специализированная библиотека для Python pyTelegramBotAPI (модуль telebot).

```
pip install pytelegrambotapi
```

1. Подключение к API с указанием токена. Результатом выполнения команды будет специальный объект, который позволит взаимодействовать с ботом.

`<имя объекта – бота> = telebot.TeleBot(<строка – токен>)`

2. Функция, которая прослушивает и обрабатывает сообщения от пользователя, поступившие на канал бота. Чтобы эта функция работала, перед ее определением необходимо поместить **декоратор**

`@<имя объекта – бота>.message_handler(content_types=["text"])`

3. Запуск цикла обработки сообщений:

`<имя объекта – бота>.polling()`

Пример

```
import telebot
```

```
bot = telebot.TeleBot("5928923473:AAEUoJIRO392cLW1vPVw57rjS0LUsMnst8c")
```

```
@bot.message_handler(content_types=['text'])
```

```
def get_text_messages(message):
```

```
    if message.text == "Привет":
```

```
        bot.send_message(message.from_user.id, "Привет! Как тебя зовут?")
```

```
        bot.register_next_step_handler(message, get_name);
```

```
    elif message.text == "/help":
```

```
        bot.send_message(message.from_user.id, "Напиши \"Привет\".")
```

```
    else:
```

```
        bot.send_message(message.from_user.id, "Не понимаю. Напиши /help.")
```

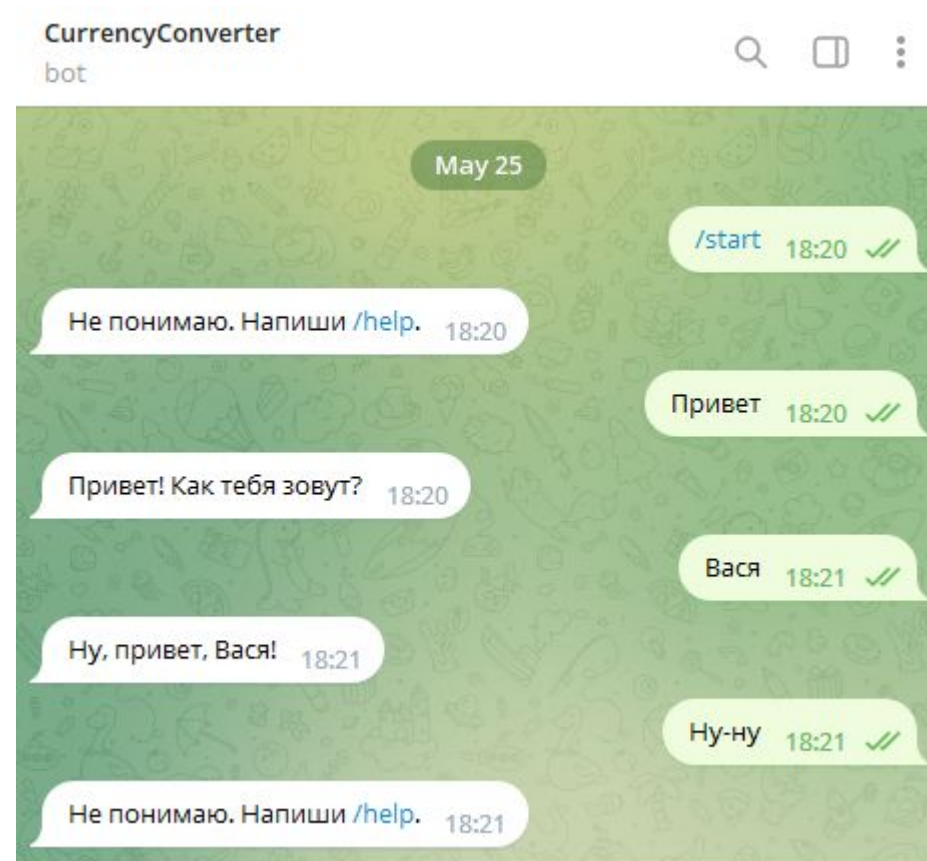
```
def get_name(message):
```

```
    global name;
```

```
    name = message.text;
```

```
    bot.send_message(message.from_user.id, f"Ну, привет, {name}!");
```

```
bot.polling()
```



Задание

- Написать Telegram-бота для перевода указанной пользователем суммы в рублях в евро и доллары. Курс перевода получать с использованием web-сервиса.

