**Урок 5**

**Создание проекта «Гороскоп». Работа с массивами**

На предыдущем уроке мы создали бота, который отправлял информацию о пользователе. Давайте вспомним, где хранится информация о пользователе, его сообщении и чате с ним? В параметре **message.** Как получить из этого параметра текст сообщения? Дописать значение **message.text**. Как узнать имя пользователя? Для этого необходимо обратиться к **message**, затем к **from\_user** и указать значение **first-name**.

Сегодня мы создадим бота с гороскопами для Telegram, научим его генерировать «предсказания» на основе фраз-шаблонов. Итак, наш бот должен:

* Здороваться по имени (если оно указано в профиле).
* Анализировать сообщения от пользователя.
* Предлагать выбрать знак зодиака.
* Генерировать «предсказания» случайным образом на основе фраз-шаблонов из массивов

Выглядит объемно, но на практике мы поэтапно разберёмся в алгоритме создания программы бота.

**Алгоритм гороскоп-бота для Telegram**

Для лучшего понимания разберёмся, как будет генерироваться «прогноз». Он будет состоять из четырёх частей:

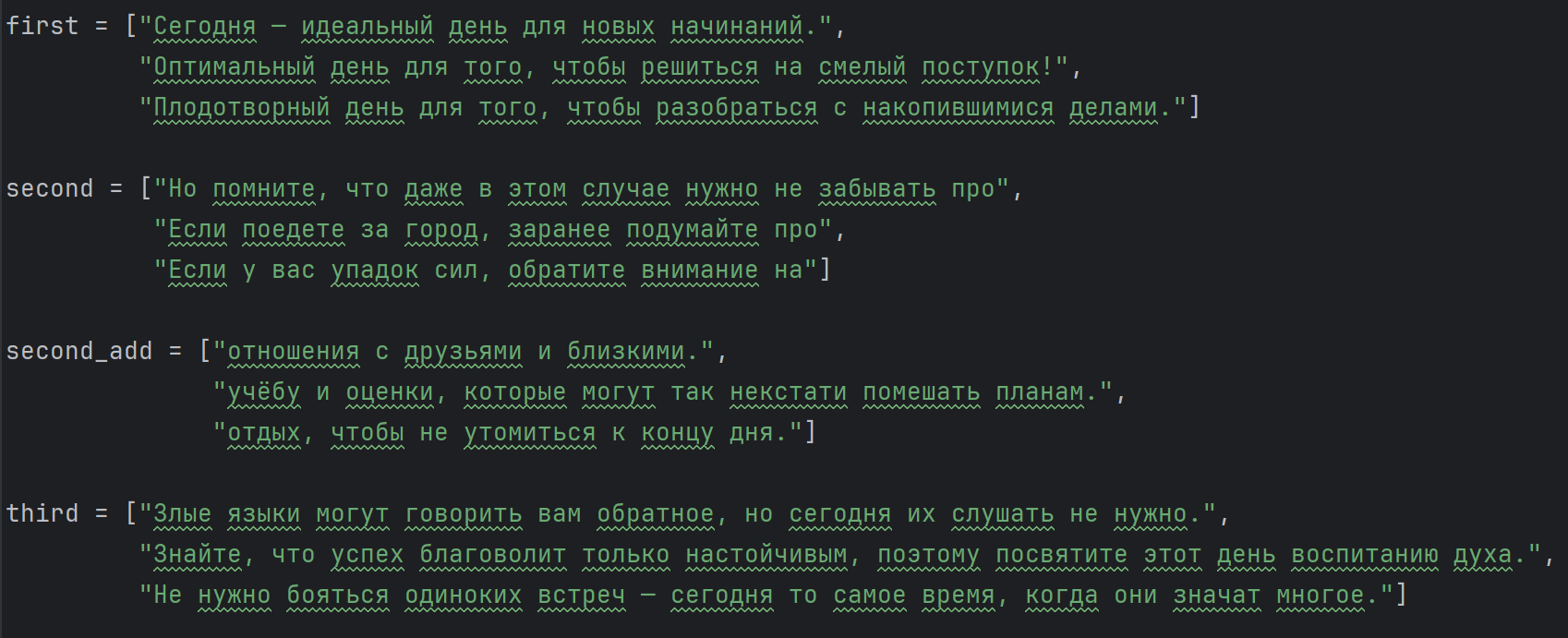
* Первая часть — предложение, задающее тон прогноза.
* Вторая часть — условие для потенциально возможной ситуации.
* Третья часть — решение возможной ситуации.
* Четвертая часть — предложение с еще одной ситуацией и ее решением.

Задача этого этапа — написать текстовые заготовки, из которых будет собираться наш гороскоп. Главная сложность в том, чтобы сделать их связными по звучанию между собой, в каком порядке они бы ни были. Вы можете использовать готовый текст из файла, прикреплённого к уроку.

Все эти части в случайном порядке перемешиваются между собой в полноценное «повествование». В примере использовано по 3 варианта фраз-шаблонов, но вы всегда сможете добавить в массивы столько шаблонов, сколько вам будет нужно. Чем их больше — тем ниже вероятность, что один и тот же прогноз продублируется.

**Самостоятельная работа**

Создайте четыре массива с фразами-шаблонами

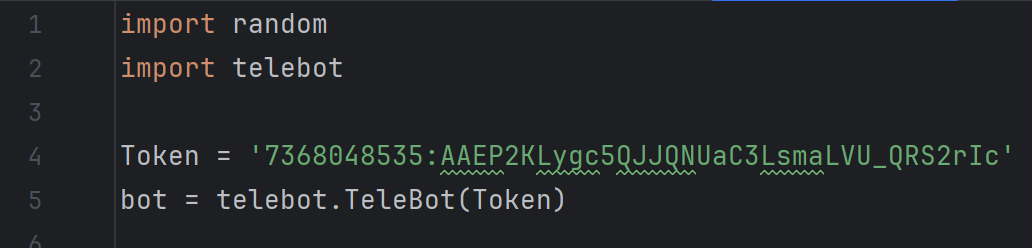


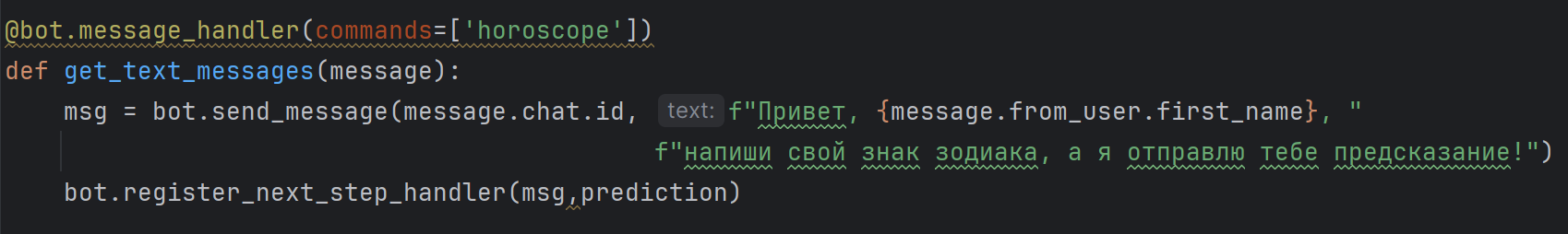
Теперь, когда мы поняли принцип работы алгоритма бота-гороскопа — перейдем к его созданию.

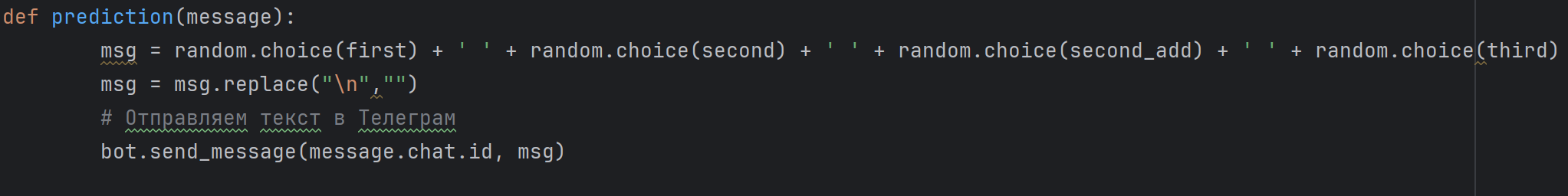
**Начинаем писать код**

Для начала необходимо добавить новую команду нашему боту, например, **/horoscope**.

В самом начале программы подключим модуль, который отвечает за случайные события — **random**. Именно с его помощью мы будем выбирать, какие предложения показывать в готовом тексте.



Создаём обработчик команды **/horoscope.** Делаем так, чтобы бот здоровался с пользователем, называя имя, указанное в профиле, и ожидал сообщение.

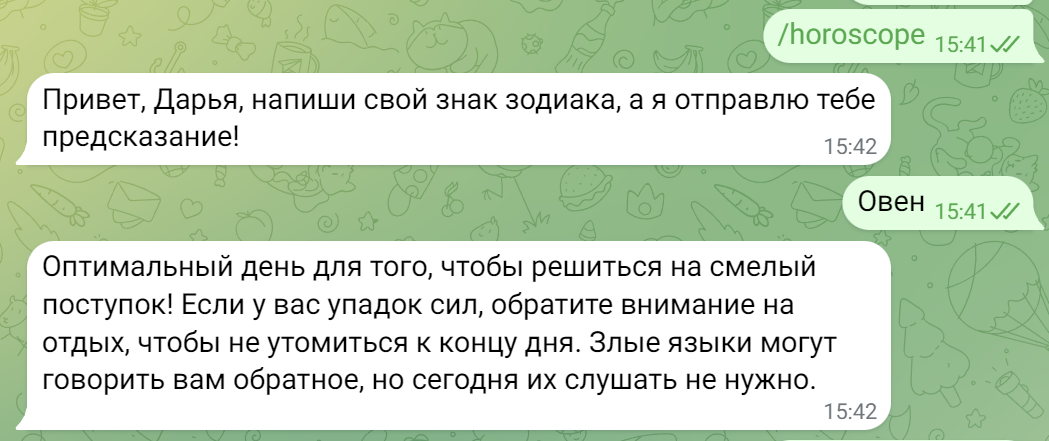
Создаём вторую функцию.

В переменную **msg** мы сохраняем готовое предсказание, состоящее из четырёх частей. Команда **random.choice()** — возвращает случайный элемент из списка (массива). В нашем случае мы положим в неё названия наших переменных с предложениями, и она случайным образом выберет их оттуда.

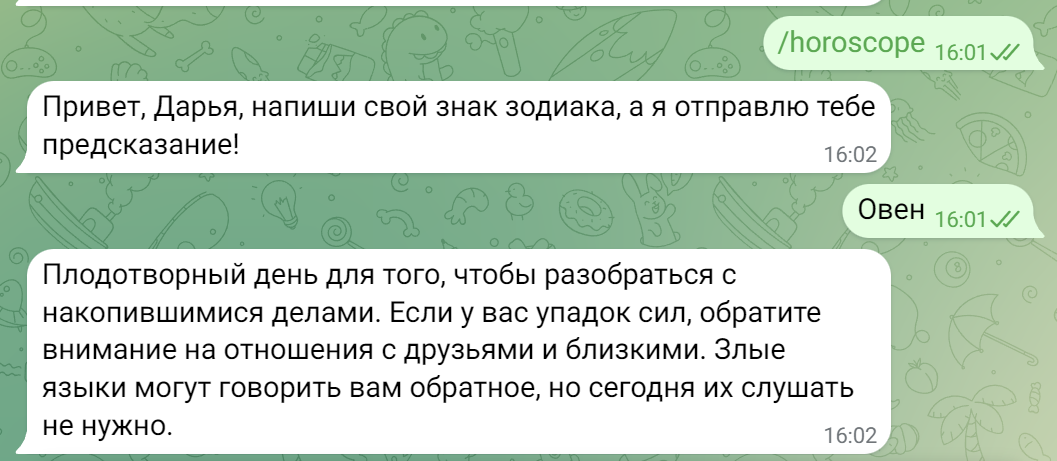
В следующей строке мы использовали команду **msg.replace,** чтобы убрать абзацы в нашем предсказании. Метод **replace** осуществляет поиск и замену в строке. В первом параметре метода указываем подстроку, которую хотим заменить. Во втором параметре - чем хотим ее заменить. Мы заменили символ «\n» на пустые кавычки «».

Вся программа «Гороскоп-бота» выглядит так:

Теперь протестируем нашего бота.



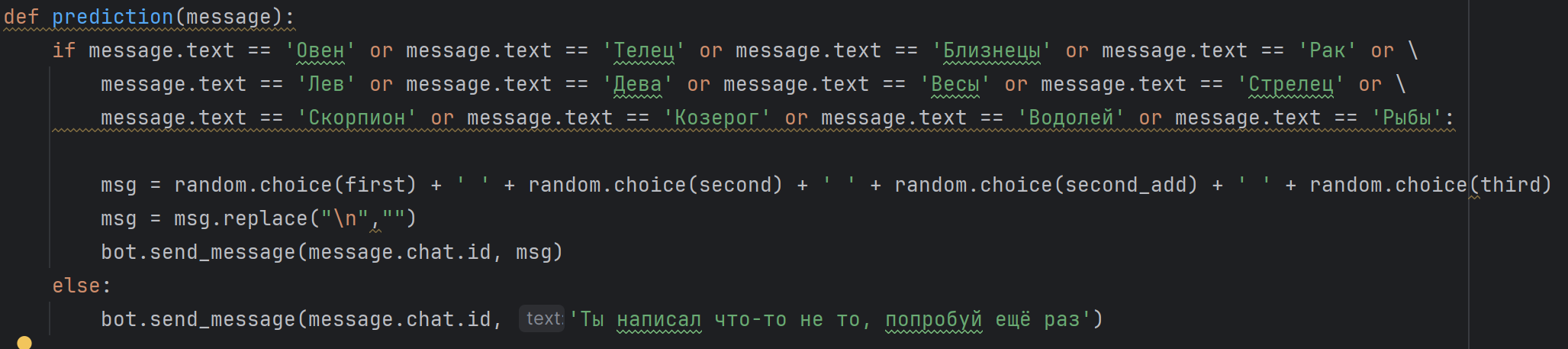
Запустим ещё раз



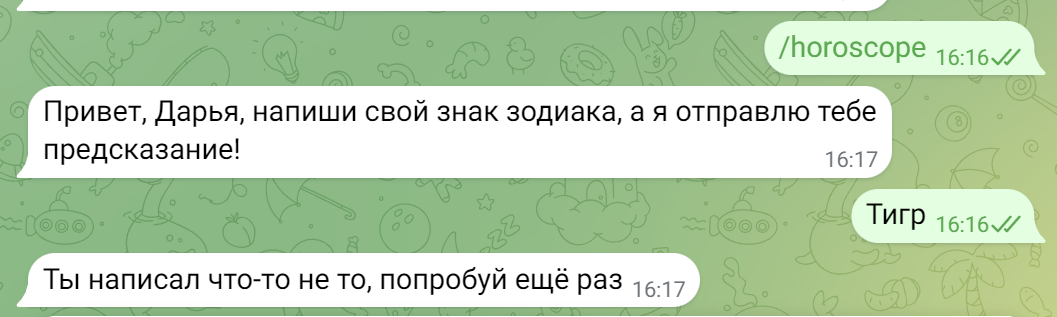
Каждый раз мы будем получать новое предсказание, которое программа сформирует случайным образом.

**Самостоятельная работа**

Также можно дополнить программу бота так, чтобы пользователь получал предсказание только в случае, если правильно указал знак зодиака. Для этого необходимо редактировать вторую функцию — добавить условие:



Теперь если пользователь напишет несуществующий знак зодиака, бот не будет выдавать предсказание.



**Рефлексия**

* Сегодня мы создали бота с гороскопами для Telegram
* Научили его генерировать «предсказания» на основе фраз-шаблонов
* Повторили как можно хранить и использовать информацию в массиве
* Использовали библиотеку random для случайного выбора фразы

***Тайминг.***

|  |  |  |
| --- | --- | --- |
|  | Раздел урока | Время с начала урока, мин. |
|  | Повторение понятия message. Постановка цели урока | 0 |
|  | Алгоритм гороскоп-бота для Telegram | 10 |
|  | Самостоятельная работа | 20 |
|  | Начинаем писать код | 40 |
|  | Тестируем бота | 60 |
|  | Бескомпьютерная деятельность | 65 |
|  | Самостоятельная работа | 70 |
|  | Рефлексия | 85 |