

Цифриум

Технологическая карта

# Умный код на Python. Базовый уровень

Модуль 4

Тема 3

Урок 6

**Создание сценариев диалогов и ботов:  
основы, лучшие практики**



### Уважаемый коллега, обратите внимание!

Учёт присутствия ребёнка на уроке ведётся только по цифровому следу. Каждый ученик из группы на уроке должен передать цифровой след на нашей платформе.

Вы отвечаете за передачу следа учеником на нашей платформе во время урока.

Передавать цифровой след нужно в личном кабинете ученика.

Цифровой след засчитывается по итогу выполнения учеником трёх задач на платформе в ходе урока.

Обращаем внимание – в задачах с отправкой ответа в виде файла наша платформа поддерживает следующие форматы: .pdf, .doc, .png, .jpeg, .xlsx, .mp3, .py, .ipynb, .txt, .csv, .json, .xml, .sb3, .ino, .hex.

Контролируйте время урока, чтобы не пропустить момент, когда ученики должны залогиниться в личном кабинете и перейти к решению задач на платформе.

Проследите, чтобы каждый ученик залогинился в личном кабинете и выполнил 3 задачи на платформе. В противном случае, Вам не будет засчитано, что вы провели урок, занятие не будет оплачено, а ученику не будет выставлена отметка о присутствии на уроке.

Просим вас в ходе 45 минут урока проконтролировать, чтобы:

1. Каждый из учеников залогинился в личном кабинете.
2. Каждый ученик в разделе **«Решить задания»** отправил решения по 3 задачам – «Практическое задание 1», «Практическое задание 2» и «Практическое задание 3».

# О занятии

### Краткое содержание занятия:

1. Узнаете, как строить диалоги для чат-ботов.
2. Изучите структуру диалога (system, user, assistant).
3. Рассмотрите подходы FSM vs LLM.
4. Освоите обработку ошибок (fallback, reprompt).
5. Проведёте практическую разработку сценария.
6. Научитесь создавать простые чат-боты на базе GPT.

### Цель:

Научить проектировать многопользовательские сценарии диалогов и создавать простые чат-боты на базе GPT.

### Задачи:

1. Изучить структуру диалога: роли (system, user, assistant) и порядок сообщений.
2. Рассмотреть подходы FSM vs LLM: когда использовать скрипты, а когда полагаться на GPT.
3. Освоить реализацию памяти и обработку ошибок (fallback, reprompt).
4. Провести практическую разработку: спроектировать диалоговый сценарий.

### Термины

1. **FSM** – сценарий по шагам, без гибкости.
2. **LLM** – ИИ вроде GPT, понимает суть и контекст.
3. **Заглушка (fallback)** – ответ, если бот не понял вопрос.
4. **Этика** – уважительное и понятное общение, даже если вы пишете код.
5. **system** – Роль – инструкции для бота.
6. **user** – Роль – сообщения пользователя.
7. **assistant** – Роль – ответы бота.

### Необходимые материалы:

Доступ к платформе для выполнения практических заданий.

## Темы и время

ЭТАП	ВРЕМЯ
Приветствие	5 мин.
Теория	25 мин.
Итоги занятия	5 мин.
Получение фидбека. Рефлексия. Проверка знаний. Домашнее задание	5 мин.
Вопрос-ответ	5 мин.
Итого	45 мин.

## Ход занятия

Номер слайда	Пояснение к слайду
	<p>Приветствие: Добро пожаловать на занятие! Сегодня мы научимся создавать сценарии диалогов для чат-ботов и изучим, как проектировать разговорные интерфейсы. Вы узнаете, как превратить набор ответов в осмысленный диалог, который будет полезен и понятен пользователю.</p> <p>Дополнительно: Чат-бот – это не просто «отвечалка на вопросы». Это интерфейс, который должен вести пользователя к цели: найти информацию, решить проблему, получить поддержку. Сегодня вы освоите принципы, по которым строятся боты в поддержке, образовании и сервисах.</p>
	<p>Титульный слайд. Тема урока: «Создание сценариев диалогов и ботов: основы, лучшие практики».</p>
1.	<p>Краткое повторение:</p> <p>Поняли, как управлять стилем и поведением модели через контекст и ролевые инструкции</p> <p>Исследовали влияние разных ролей (эксперт, блогер, учитель) на тональность ответа</p> <p>Сравнили три стиля подачи одного запроса и оценили их уместность для разных задач</p> <p>Научились формулировать инструкции для точной настройки стиля ответа</p> <p>Провели практическую работу с разными ролями и зафиксировали результаты</p>

## Ход занятия

	<p>Усвоили: стиль зависит от цели задачи, аудитории и формата</p> <p>Дополнительно: Вы запомнили: один и тот же факт можно подать по-разному. Сегодня мы переходим на новый уровень: как выстроить не один ответ, а целый диалог, где каждый шаг ведёт пользователя к результату.</p>
2.	<p>Сегодня вы узнаете:</p> <p>Как строить диалоги для чат-ботов: от приветствия до завершения</p> <p>Структуру диалога с ролями: system, user, assistant – и как они взаимодействуют</p> <p>Два подхода к управлению диалогом: FSM (скрипты) vs LLM (свободная генерация)</p> <p>Как обрабатывать ошибки: fallback-ответы, reprompt, валидация ввода</p> <p>Проведёте практику: спроектируете сценарий для школьного бота-помощника</p> <p>Научитесь создавать простые чат-боты на базе GPT с учётом этики и удобства</p> <p>Дополнительно: К концу урока у вас будет готовый шаблон сценария, который можно адаптировать под любую задачу: библиотека, расписание, помощь с учёбой. Это основа для ваших будущих проектов.</p>
3.	<p>Цель: Научить проектировать многопользовательские сценарии диалогов и создавать простые чат-боты на базе GPT.</p>

## Ход занятия

	<p>Задачи:</p> <p>Изучить структуру диалога с ролями system, user, assistant и порядок сообщений для построения контекста</p> <p>Рассмотреть два подхода к управлению диалогом: FSM (конечный автомат) и LLM (языковая модель) – когда использовать скрипты, а когда довериться ИИ</p> <p>Освоить реализацию «памяти» диалога и обработку ошибок: fallback, reprompt, валидация</p> <p>Спроектировать диалоговый сценарий для конкретной задачи с учётом пользовательского пути</p> <p>Дополнительно: После урока вы сможете отличить хорошо спроектированный диалог от хаотичного и самостоятельно создавать сценарии, которые действительно помогают пользователям, а не запутывают их.</p>
4.	<p>Ключевые понятия. FSM (Finite State Machine, конечный автомат) – подход, при котором диалог идёт по заранее прописанным шагам (состояниям). Пользователь выбирает из вариантов, бот переходит к следующему шагу.</p> <p>Плюсы: предсказуемо, легко тестировать, контроль над сценарием.</p> <p>Минусы: негибко, ломается при нестандартных запросах.</p> <p>LLM (Large Language Model) – подход, при котором ИИ понимает суть запроса и генерирует ответ в контексте диалога.</p> <p>Плюсы: гибко, понимает естественный язык, адаптируется.</p> <p>Минусы: может «галлюцинировать», сложнее контролировать, требует больше токенов.</p> <p>Fallback (заглушка) – ответ, который бот даёт, если не распознал намерение пользователя.</p> <p>Пример: «Извините, я не понял вопрос. Попробуйте переформулировать или выберите из меню: [варианты]».</p>

## Ход занятия

	<p>Этика в диалогах – принципы уважительного и понятного общения: бот должен быть вежливым, прозрачным («я – бот»), признавать ошибки и не требовать лишней информации.</p> <p>Дополнительно: FSM и LLM не конкурируют – их можно комбинировать. Например, FSM для навигации по меню, LLM для ответов на свободные вопросы.</p>
5.	<p>FSM (скриптовый подход):</p> <p>Подходит для: простых задач с чёткими шагами (заказ пиццы, запись к врачу, тест с вариантами)</p> <p>Плюсы: полный контроль, легко отлаживать, предсказуемый расход токенов</p> <p>Минусы: не понимает перефразировки, ломается при отклонении от сценария</p> <p>LLM (подход на основе ИИ):</p> <p>Подходит для: сложных диалогов, где пользователь формулирует запрос свободно (консультация, обучение, творчество)</p> <p>Плюсы: понимает контекст, адаптируется под стиль пользователя, гибкий</p> <p>Минусы: может ошибаться, сложнее тестировать, требует мониторинга</p> <p>Гибридный подход (лучшее из двух):</p> <p>FSM для навигации: «Выберите: 1) Расписание, 2) Библиотека, 3) Помощь»</p> <p>LLM для ответов: «Расскажи подробнее про книги по истории для 9 класса»</p>

## Ход занятия

	<p>Дополнительно: Начинайте с FSM для критических путей (где ошибка недопустима), подключайте LLM там, где нужна гибкость. Тестируйте оба подхода на реальных запросах.</p>
6.	<p>Три ключевые роли в диалоге с ИИ:</p> <p>system – инструкции для бота, задающие поведение и контекст</p> <p>Пример: «Ты – помощник школьной библиотеки. Отвечай кратко, вежливо, предлагай книги по возрасту. Если не знаешь – честно говори.»</p> <p>user – сообщения от пользователя (вопросы, запросы, выборы)</p> <p>Пример: «Какие книги по истории есть для 9 класса?»</p> <p>assistant – ответы бота (генерируются моделью или выбираются из скрипта)</p> <p>Пример: «Для 9 класса рекомендую: "История России" автор Иванов, "Всемирная история" автор Петрова. Хочешь, расскажу подробнее о любой из них?»</p> <p>Порядок сообщений важен:</p> <p>Модель обрабатывает массив messages сверху вниз</p> <p>История диалога должна идти в хронологическом порядке</p> <p>system-сообщение обычно идёт первым и задаёт тон всему диалогу</p> <p>Дополнительно: Для «памяти» диалога сохраняйте историю messages и передавайте её в каждом новом запросе. Без этого модель «забудет», о чём вы говорили ранее.</p>

## Ход занятия

7.	<p>Что делать, если бот не понял пользователя:</p> <p>Fallback (заглушка) – безопасный ответ при нераспознанном запросе:</p> <p>«Извините, я не понял вопрос. Попробуйте переформулировать.»</p> <p>«Вот что я умею: [список возможностей]. Выберите или задайте вопрос иначе.»</p> <p>Важно: не извиняться многократно, а сразу предлагать решение</p> <p>Reprompt (повторный запрос) – возврат к предыдущему шагу:</p> <p>Если пользователь дал некорректный ввод: «Пожалуйста, введите номер класса от 1 до 11»</p> <p>Если запрос неполный: «Уточните, по какому предмету нужны книги?»</p> <p>Валидация ввода – проверка данных до обработки:</p> <p>Числа: «Введите номер от 1 до 11»</p> <p>Выбор из списка: «Выберите: 1) История, 2) Математика, 3) Литература»</p> <p>Формат: «Введите дату в формате ДД.ММ.ГГГГ»</p> <p>Дополнительно: Хороший fallback не просто говорит «не понял», а направляет пользователя: предлагает варианты, подсказывает формат, возвращает к безопасному пути.</p>
----	--

## Ход занятия

8.	<p>Этика в чат-ботах — это принципы, которые делают взаимодействие с ботом безопасным, уважительным и полезным для пользователя.</p> <p>Правило 1: Вежливость и уважительный тон</p> <p>Бот использует корректные формы обращения и благодарности</p> <p>Сохраняет нейтральный или позитивный тон даже при повторяющихся ошибках пользователя</p> <p>Адаптирует стиль общения под целевую аудиторию и контекст задачи</p> <p>Избегает снисходительности, агрессии и пассивного сопротивления</p> <p>Почему важно: Вежливость формирует доверие и повышает готовность пользователя продолжать диалог.</p> <p>Правило 2: Понятность для пользователя</p> <p>Бот использует язык, соответствующий уровню знаний целевой аудитории</p> <p>Структурирует информацию: логическая последовательность, разбивка на части, выделение главного</p> <p>Даёт обратную связь о процессе: подтверждение получения запроса, индикация выполнения</p> <p>Избегает избыточного жаргона или поясняет специальные термины при необходимости</p> <p>Почему важно: Понятный ответ = полезный ответ. Если пользователь не понимает, он не получает ценность.</p>
----	--

## Ход занятия

	<p>Правило 3: Честность и признание ошибок</p> <p>Бот не генерирует информацию, в которой не уверен</p> <p>Открыто сообщает об ограничениях: «У меня нет этих данных», «Попробуйте переформулировать»</p> <p>Предлагает альтернативные пути решения вместо тупика</p> <p>Формулирует ошибки как свою ответственность, а не как вину пользователя</p> <p>Почему важно: Честность сохраняет доверие. Пользователь ценит прозрачность больше, чем иллюзию всезнания.</p> <p>Правило 4: Минимализм данных – не требовать лишнего</p> <p>Бот запрашивает только те данные, которые необходимы для решения конкретной задачи</p> <p>Объясняет цель сбора информации, если запрос может вызвать вопросы</p> <p>Предоставляет возможность пропустить необязательные поля</p> <p>Не сохраняет и не использует личные данные без явного согласия</p> <p>Почему важно: Избыточный сбор данных нарушает приватность, снижает доверие и создаёт риски безопасности.</p>
9.	<p>Практика: Задание: Спроектируйте диалоговый сценарий для школьного бота-помощника.</p> <p>Этапы сценария:</p>

## Ход занятия

	<p>Приветствие: «Привет! Я – школьный помощник. Чем могу помочь?»</p> <p>Навигация: предложить варианты (расписание, библиотека, вопросы по учёбе, контакты)</p> <p>Обработка 3 типов запросов:</p> <p>Факт: «Когда контрольная по математике?» → ответ с датой</p> <p>Рекомендация: «Какие книги почитать по истории?» → список с кратким описанием</p> <p>Инструкция: «Как записаться в кружок?» → пошаговое руководство</p> <p>Заглушка при ошибке: «Не понял вопрос. Попробуйте: "расписание", "книги", "кружки"»</p> <p>Завершение: «Спасибо! Обращайся, если что-то нужно.»</p> <p>Инструкция:</p> <p>Запишите сценарий в виде таблицы: шаг → ожидаемый ввод пользователя → ответ бота</p> <p>Продумайте 1–2 варианта fallback для нестандартных запросов</p> <p>Добавьте system-инструкцию для модели (если используете LLM)</p> <p>Дополнительно: Протестируйте сценарий «в уме»: представьте, что вы пользователь с разным уровнем знаний. Поймёт ли новичок? Не запутается ли опытный?</p>
10.	Пример: Бот-помощник библиотеки

## Ход занятия

system: «Ты – помощник школьной библиотеки. Отвечай кратко, вежливо, предлагай книги по возрасту и интересам. Если не знаешь – честно говори.»

Приветствие:

Бот: «Привет! Я – библиотечный помощник 📖 Чем могу помочь?»

Вопрос пользователя:

Пользователь: «Какие книги по истории есть для 9 класса?»

Ответ бота:

Бот: «Для 9 класса рекомендую:

- "История России" (авт. Иванов) – обзор ключевых событий
- "Всемирная история" (авт. Петрова) – с картами и хроникой

Хочешь, расскажу подробнее о любой из них или подберу по конкретной теме?»

Заглушка (если запрос непонятен):

Бот: «Извините, не совсем понял. Попробуйте: "книги по истории", "новые поступления", "как записаться в библиотеку"»

Завершение:

Бот: «Спасибо, что обратился! Обращайся ещё 📖»

Принципы этики в ботах:

Вежливость: «пожалуйста», «спасибо», без сленга, если не задан дружеский тон

## Ход занятия

	<p>Прозрачность: сообщать, что пользователь общается с ботом</p> <p>Честность: признавать, если бот не знает ответа, вместо выдумки</p> <p>Минимализм данных: не запрашивать лишнюю личную информацию</p> <p>Доступность: простой язык, понятные формулировки, альтернативы для сложных терминов</p> <p>Дополнительно: Этика – не «довесок», а основа доверия. Пользователь простит ошибку, но не простит неуважение или обман.</p>
--	---

## Практика

### Задание 1

Ситуация: Бот помогает ученику узнать полезную информацию. Он:

Приветствует

Задаёт вопрос «Чем помочь?»

Отвечает на 3 запроса по выбору

Отвечает заглушкой при ошибке

Завершает разговор

Этап	Реплика бота
Приветствие	
Вопрос	
Ответ 1	
Ответ 2	
Ответ 3	
Заглушка	
Завершение	

### Задание 2

Нарисуйте блок-схему диалога (текстовое описание):

## Примеры выполнения заданий

### Задание 1.

Этап	Реплика бота
Приветствие	Привет! Я твой помощник по учёбе
Вопрос	Чем могу помочь?
Ответ 1	Конечно! Вот кратко: фотосинтез – это процесс, при котором растения превращают солнечный свет, воду и углекислый газ в кислород и глюкозу. Происходит в листьях, благодаря хлорофиллу.
Ответ 2	Площадь прямоугольника считается так: длина $\times$ ширина. Например, если длина 5 м, а ширина 3 м, то площадь = 15 м <sup>2</sup> .
Ответ 3	В «Войне и мире» Лев Толстой показывает жизнь русского общества во время Наполеоновских войн. Главные герои – Пьер Безухов, Андрей Болконский и Наташа Ростова. Роман учит милосердию, поиску смысла и силе духа.
Заглушка	Не уверен, что понял. Можешь переформулировать вопрос?
Завершение	Удачи в учёбе! Если что – я всегда рядом

## Примеры выполнения заданий

### Задание 2

[Начало]

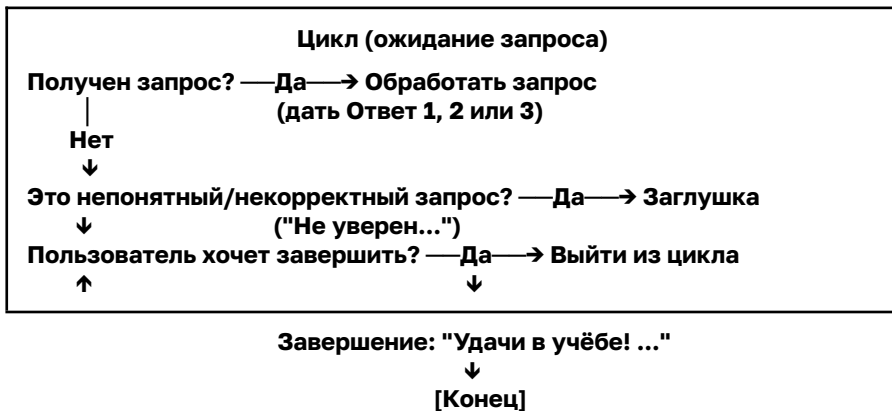
↓

Приветствие: "Привет! Я твой помощник по учёбе. 😊"

↓

Вопрос: "Чем могу помочь?"

↓



### Сегодня на занятии мы:

- Научились проектировать сценарии диалогов для чат-ботов.
- Изучили структуру диалога: роли (system, user, assistant) и порядок сообщений.
- Рассмотрели подходы FSM vs LLM: когда использовать скрипты, а когда GPT.
- Освоили реализацию памяти и обработку ошибок (fallback, reprompt).
- Провели практическую разработку: спроектировали диалоговый сценарий.
- Узнали о важности этики в создании ботов.

### На следующем занятии вы:

- Подведёте итоги всех пройденных модулей по GPT и нейросетям.
- Обсудите, чему научились и где применимы навыки работы с GPT.
- Проведёте анализ вызовов: этика, приватность, надёжность.
- Познакомитесь с успешными и неудачными кейсами применения GPT.

### Получение фидбека. Рефлексия. Проверка знаний

Попросите ребят решить интерактивные задачи по теме урока.

## Ответы к интерактивным задачам

### Практическое задание 1

Что такое FSM (Finite State Machine) в контексте создания чат-ботов?

- **Сценарий диалога по заранее определённым шагам, без гибкости**
- Искусственный интеллект, понимающий любой контекст
- База данных для хранения сообщений
- Протокол передачи данных между ботом и сервером

### Практическое задание 2

Какие элементы необходимы для создания сценария простого чат-бота?

- **Приветствие**
- **Заглушка (fallback) на случай ошибки**
- **Завершение разговора**
- Графический интерфейс с кнопками
- Подключение к социальным сетям

### Практическое задание 3

Как называется ответ бота, когда он не понял вопрос пользователя? (одно слово на английском)

**fallback**

### Вопрос-ответ

Если во время подготовки или проведения урока возникают вопросы, их необходимо адресовать в методический чат.