

1 ЧАСТЬ

SQLALCHEMY

Продвинутый Python

Андрей Вильмов

СОДЕРЖАНИЕ ТЕМЫ

Продвинутый Python

01 ORM

02 SQLAlchemy

03 Создание таблиц и подключение к базе данных

ORM

Object-Relational Mapping

технология, связывающая объектную
модель приложения с реляционной базой
данных для упрощения работы с данными

SQLALCHEMY

это инструмент для работы с базами данных
в Python, обеспечивающий ORM-функциональность
и возможность создания SQL-запросов

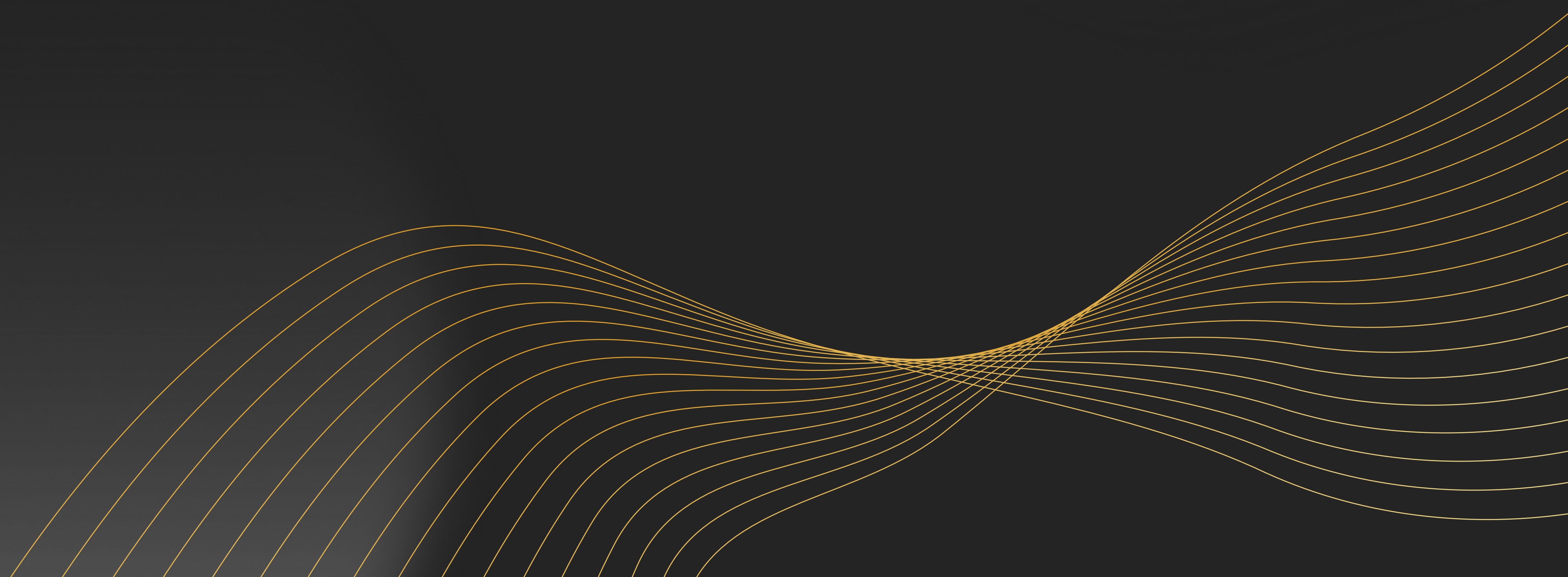
ПОДКЛЮЧЕНИЕ К БАЗЕ ДАННЫХ

Продвинутый Python

Для подключения к базе данных
используется **create_engine**

...

```
connection_string = f"postgresql://postgres:password@localhost:5432/synergy"  
engine = create_engine(connection_string)
```



SESSIONMAKER

Это фабрика объектов "сессии"
в библиотеке SQLAlchemy, обеспечивающая
удобный доступ к базе данных через ORM

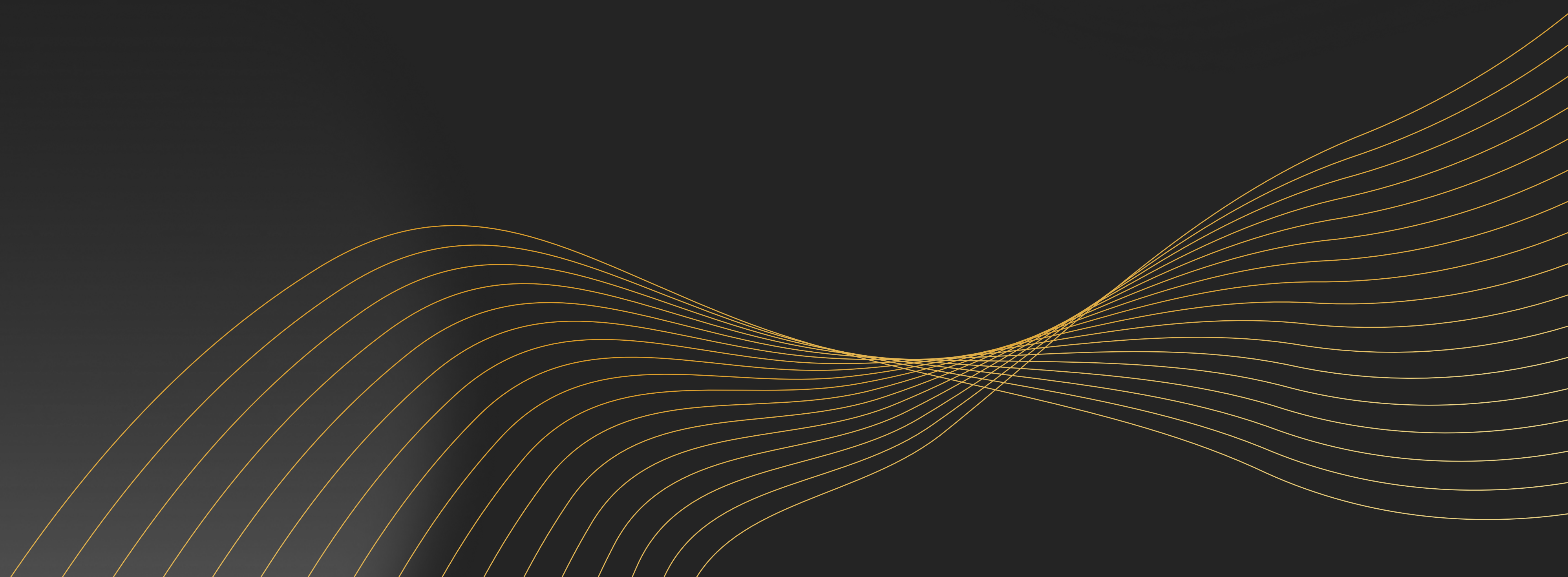
СОЗДАНИЕ СЕССИИ

Продвинутый Python

Для создания сессии используется **sessionmaker**

...

```
Session = sessionmaker(bind=engine)  
Session = Session()
```



DB.TABLE

это класс в библиотеке SQLAlchemy,
представляющий таблицу в базе данных
и используемый для определения её структуры

СОЗДАНИЕ ТАБЛИЦ

Продвинутый Python

Для создания таблиц используется **db.Table** в котором описываются все поля таблицы с типами этих полей

```
...  
  
profile = db.Table(  
    'profile',  
    metadata_obj,  
    db.Column('email', db.String, primary_key=True),  
    db.Column('name', db.String),  
    db.Column('contact', db.Integer),  
)
```


ПОДВЕДЁМ ИТОГИ

Продвинутый Python

- ✓ ORM помогает забыть про СУБД которая стоит на бэк-энде
- ✓ ORM позволяет сразу проверять и создавать таблицы в базе
- ✓ При обращении и работе с базой стоит использовать сессии