

Adding Authentication



Reindert-Jan Ekker

@rjekker www.codesensei.nl



Overview



Adding a User Class

- DB Options: unique, index
- Password hashing

HTTP Basic Auth

- Nested dependencies

OAuth 2

Extra Model Field Options

```
from sqlalchemy import SQLModel, Field, Column, VARCHAR

class User(SQLModel, table=True):
    id: int | None = Field(default=None, primary_key=True)
    username: str = Field(
        sa_column=Column("username", VARCHAR, unique=True, index=True))
    password_hash: str = ""
```



Password Hashing

```
from passlib.context import CryptContext

pwd_context = CryptContext(schemes=["bcrypt"])

class User(SQLModel, table=True):
    ...
    password_hash: str = ""

    def set_password(self, password):
        self.password_hash = pwd_context.hash(password)

    def verify_password(self, password):
        return pwd_context.verify(password, self.password_hash)
```



Links



<https://bit.ly/fastapi-oauth> (includes JWT)



<https://bit.ly/fastapi-oauth-scopes>



Summary



Adding a User Class

- DB Options: unique, index
- Password hashing

HTTP Basic Auth

OAuth 2



Up Next: Testing and Deploying

