R.S.KARTHIKEYAN
19/08/2025

CODE

```
from sqlalchemy import create_engine, Column, Integer, String
from sqlalchemy.orm import sessionmaker, declarative_base, Session
from fastapi import FastAPI, Depends, Request, HTTPException
from typing import List
import uvicorn
from pydantic import BaseModel

SQLACHEMY_DATABASE_URL = "sqlite:///./MusicList.db"

engine = create_engine(SQLACHEMY_DATABASE_URL,
connect_args={"check_same_thread": False})
SessionLocal = sessionmaker(autocommit=False, autoflush=False,
bind=engine)
Base = declarative_base()

class Music(Base):
    __tablename__ = "music"
    Id = Column(Integer, primary_key=True, index=True)
    Name = Column(String(50), nullable=False)
    MusicDirector = Column(String(50), nullable=False)
    Album = Column(String(50), nullable=False)

class User(Base):
    __tablename__ = "user"
    Id = Column(Integer, primary_key=True, index=True)
    Name = Column(String(50), nullable=False)
    Age = Column(Integer, nullable=False)


class Musics(BaseModel):
    Id: int
    Name: str
    MusicDirector: str
    Album: str
    class Config:
        orm_mode = True
```

R.S.KARTHIKEYAN
19/08/2025

```python
class MusicInput(BaseModel):
    Name: str
    MusicDirector: str
    Album: str

class UserInput(BaseModel):
    Name: str
    Age: int

class Music_dto(BaseModel):
    Name: str
    MusicDirector: str
    Album: str
    class Config:
        orm_mode = True
        from_attributes = True

app = FastAPI()

def get_db():
    db = SessionLocal()
    try:
        yield db
    finally:
        db.close()

# Endpoints
@app.post('/add_music')
def add_music(st1: MusicInput, db: Session = Depends(get_db)):
    st = Music(Name=st1.Name, MusicDirector=st1.MusicDirector,
Album=st1.Album)
    db.add(st)
    db.commit()
    db.refresh(st)
    return {"message": "Music Inserted"}

@app.post('/addUser')
def add_user(usr: UserInput, db: Session = Depends(get_db)):
    user = User(Name=usr.Name, Age=usr.Age)
    db.add(user)
```

R.S.KARTHIKEYAN
19/08/2025

```python
    db.commit()
    db.refresh(user)
    return {"message": "User Inserted"}

def validate_user(UserId: int, db: Session = Depends(get_db)):
    user = db.query(User).filter_by(Id=UserId).first()
    if not user:
        raise HTTPException(status_code=404, detail="User not found")
    return UserId

@app.get('/GetMusics/{UserId}', response_model=List[Musics])
def get_musics(UserId: int = Depends(validate_user), db: Session =
Depends(get_db)):
    return db.query(Music).all()

@app.get('/GetUsers')
def get_users(db: Session = Depends(get_db)):
    return db.query(User).all()

@app.get('/GetbyName/{Music_Name}', response_model=Music_dto)
def get_by_name(Music_Name: str, db: Session = Depends(get_db)):
    music = db.query(Music).filter(Music.Name == Music_Name).first()
    if music is None:
        raise HTTPException(status_code=404, detail="Music not found")
    return Music_dto.model_validate(music)

# Middleware
@app.middleware("http")
async def addmiddleware(request: Request, call_next):
    print("Middleware Integrated")
    response = await call_next(request)
    return response

if __name__ == "__main__":
    uvicorn.run("Music:app", host="127.0.0.1", port=7000, reload=True)
```

R.S.KARTHIKEYAN
19/08/2025

OUTPUT

Request URL

```
http://127.0.0.1:7000/GetMusics/1
```

Server response

Code	Details
200	Response body


```
[
  {
    "Id": 1,
    "Name": "PowerHouse",
    "MusicDirector": "Anirudh",
    "Album": "Coolie"
  },
  {
    "Id": 2,
    "Name": "I am the Danger",
    "MusicDirector": "Anirudh",
    "Album": "Coolie"
  },
  {
    "Id": 3,
    "Name": "Naa Ready than",
    "MusicDirector": "Anirudh",
    "Album": "Leo"
  }
]
```


Download

Request URL

```
http://127.0.0.1:7000/GetMusics/9
```

Server response

Code	Details
404 *Undocumented*	Error: Not Found

Response body


```
{
  "detail": "User not found"
}
```


Download

Curl

```
curl -X 'POST' \
  'http://127.0.0.1:7000/addUser' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "Name": "Kavin",
  "Age": 22
}'
```

Request URL

```
http://127.0.0.1:7000/addUser
```

Server response

Code	Details
200	Response body


```
{
  "message": "User Inserted"
}
```


Download