

ONLINE SCHOOL WEBSITE

ZHAMBYLBAYEV AZAMAT

КТО БУДЕТ ПОЛЬЗОВАТЬСЯ?

ШКОЛЫ ПРОГРАММИРОВАНИЯ

ШКОЛЫ

ОБРАЗОВАТЕЛЬНЫЕ ЦЕНТРЫ

УНИВЕРСИТЕТЫ

КОЛЛЕДЖИ

ЯЗЫКОВЫЕ КУРСЫ

ТЕХНОЛОГИИ



КАК РАБОТАЕТ?



```
└─ Azamat
    class User(AbstractUser):
        id = models.UUIDField(primary_key=True, default=uuid.uuid4)
        username = None
        email = models.EmailField(unique=True)
        first_name = models.CharField(max_length=255)
        last_name = models.CharField(max_length=255)
        is_verified = models.BooleanField(default=False)
        otp = models.CharField(max_length=6, null=True, blank=True)
        created_at = models.DateTimeField(auto_now_add=True)
        updated_at = models.DateTimeField(auto_now=True)

        USERNAME_FIELD = 'email'
        REQUIRED_FIELDS = []

        user_type = models.CharField(
            choices=choices.UserTypeChoices.choices,
            default=choices.UserTypeChoices.Student,
            null=True,
            blank=True,
            max_length=8,
        )
    objects = UserManager()
```

```
from django.db import models
4 usages └─ Azamat
class UserTypeChoices(models.TextChoices):
    Teacher = 'Teacher'
    Student = 'Student'
```



HTTP <http://127.0.0.1:8000/api/v1/accounts/register/>

POST <http://127.0.0.1:8000/api/v1/accounts/register/>

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

Key	Value
<input checked="" type="checkbox"/> email	azambylbaev@gmail.com
<input checked="" type="checkbox"/> password	Aza404040

Key Value

Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

```
1 {  
2     "status": 200,  
3     "message": "Registration successfully check email",  
4     "data": {  
5         "email": "azambylbaev@gmail.com",  
6         "is_verified": false  
7     }  
8 }
```

Verification

Gmail Поиск в почте

Написать

Входящие 1923

Помеченные
Отложенные
Отправленные
Черновики 8
Ещё

Ярлыки +

Your account verification email Входящие x

onlajnskola95@gmail.com Кому: мне ▾
Your otp is 8055

Ответить Переслать

accounts

POST /accounts/register/

POST /accounts/token/

POST /accounts/token/refresh/

POST /accounts/token/verify/

POST /accounts/verify/

АУТЕНТИФИКАЦИЯ

```
SIMPLE_JWT = {  
    "ACCESS_TOKEN_LIFETIME": timedelta(minutes=15),  
    "REFRESH_TOKEN_LIFETIME": timedelta(days=1),  
    "ROTATE_REFRESH_TOKENS": False,  
    "BLACKLIST_AFTER_ROTATION": False,  
    "UPDATE_LAST_LOGIN": False,  
  
    "ALGORITHM": "HS256",  
    "SIGNING_KEY": SECRET_KEY,  
    "VERIFYING_KEY": "",  
    "AUDIENCE": None,  
    "ISSUER": None,  
    "JSON_ENCODER": None,  
    "JWK_URL": None,  
    "LEEWAY": 0,  
  
    "AUTH_HEADER_TYPES": ("Bearer",),  
    "AUTH_HEADER_NAME": "HTTP_AUTHORIZATION",  
    "USER_ID_FIELD": "id",  
    "USER_ID_CLAIM": "user_id",  
    "USER_AUTHENTICATION_RULE": "rest_framework_simplejwt.authentication.JWTAuthentication"}  
}
```

USER

TEACHER

STUDENT

ADMIN



```
• Azamat
class Course(models.Model):
    title = models.CharField(max_length=255)
    description = models.TextField()
    teacher = models.ForeignKey(User, on_delete=models.CASCADE)
    students = models.ManyToManyField(User, related_name='courses', blank=True)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)

    • Azamat
    def __str__(self):
        return self.title

16 usages • Azamat
class Lesson(models.Model):
    title = models.CharField(max_length=255)
    description = models.TextField()
    content = models.TextField()
    video = models.FileField(upload_to='lessons/videos/')
    course = models.ForeignKey(Course, on_delete=models.CASCADE, related_name='lessons')
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)

    • Azamat
    def __str__(self):
        return self.title
```

HTTP <http://127.0.0.1:8000/api/v1/dashboard/courses/>

GET http://127.0.0.1:8000/api/v1/dashboard/courses/

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Authorization Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJ0b...

Key Value

Body Cookies Headers (10) Test Results Status:

Pretty Raw Preview Visualize JSON ↻

```
1 {
2     "id": 1,
3     "title": "Python Programming Language"
4 },
5 {
6     "id": 2,
7     "title": "Java Programming Language"
8 },
9 {
10    "id": 3,
11    "title": "IOS development"
12 },
13 {
14    "id": 4,
15    "title": "English for beginners"
16 },
17 {
18    "id": 5,
19    "title": "Turkish for advanced"
20 },
21 }
22 ]
```

HTTP <http://127.0.0.1:8000/api/v1/dashboard/courses/1/>

GET <http://127.0.0.1:8000/api/v1/dashboard/courses/1/>

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Key	Value
Authorization	Bearer eyJhbGciOiJIUzI1NilsInR5cCI6IkpXVCJ9.eyJ0b

Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

```
1 {  
2     "id": 1,  
3     "title": "Python Programming Language",  
4     "description": "Python is a high-level, general-purpose programming language.",  
5     "teacher": {  
6         "first_name": "Saule",  
7         "last_name": "Apai",  
8         "email": "teacher1@gmail.com"  
9     }  
10 }
```

HTTP <http://127.0.0.1:8000/api/v1/dashboard/courses/1/lessons/>

GET <http://127.0.0.1:8000/api/v1/dashboard/courses/1/lessons/>

Params Authorization Headers (7) Body Pre-request Script Tests Settings

<input checked="" type="checkbox"/> Authorization	Bearer eyJhbGciOiJIUzI1NilsInR5cCI6IkpXV
Key	Value

Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

```
1
2     "lessons": [
3         {
4             "id": 1,
5             "title": "Lesson 1",
6             "description": "this is introduction"
7         },
8         {
9             "id": 2,
10            "title": "Lesson 2",
11            "description": "this is second lesson"
12        },
13        {
14            "id": 3,
15            "title": "Lesson 3",
16            "description": "this is thridlesson"
17        },
18        {
19            "id": 4,
20            "title": "Lesson 4",
21            "description": "this is fourth lesson"
22        }
23    ]
24 ]
```

HTTP <http://127.0.0.1:8000/api/v1/dashboard/courses/1/lessons/1>

GET <http://127.0.0.1:8000/api/v1/dashboard/courses/1/lessons/1>

Params Authorization Headers (7) Body Pre-request Script Tests Settings

<input checked="" type="checkbox"/> Authorization	Bearer eyJhbGciOiJIUzI1NilsInR5cCI6IkpXV
Key	Value

Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

```
1
2     "id": 1,
3     "title": "Lesson 1",
4     "description": "this is introduction",
5     "content": "tkmgopmtge",
6     "video": "/media/lessons/videos/videolesson1_A3c8JKr.mp4",
7     "created_at": "2023-07-13T17:22:42.016723Z",
8     "updated_at": "2023-07-13T17:44:47.290890Z",
9     "course": 1
10 ]
```

PUT http://127.0.0.1:8000/api/v1/my_profile/

Params Authorization Headers (9) **Body** • Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

Key	Value
<input checked="" type="checkbox"/> first_name	Azamat
<input checked="" type="checkbox"/> last_name	Zhambylbayev

Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

```
1 {  
2     "first_name": "Azamat",  
3     "last_name": "Zhambylbayev"  
4 }
```

ИНФОРМАЦИЯ

my_profile

GET /my_profile/

PUT /my_profile/

HTTP <http://127.0.0.1:8000/api/v1/dashboard/courses/1/join/>

POST <http://127.0.0.1:8000/api/v1/dashboard/courses/1/join/>

Params	Authorization	Headers (9)	Body	Pre-request Script	Tests	Settings
				,		
				(i) gzip, deflate, br		
				(i) keep-alive		
				Bearer eyJhbGciOiJIUzI1NilsInR5cCI6IkpXVCJ9.eyJ0b2...		

Body Cookies Headers (10) Test Results Status: 200 OK Time

Pretty Raw Preview Visualize JSON

```
1 {  
2   "message": "Joined the course successfully"  
3 }
```

JOIN K KYPCAM

HTTP http://127.0.0.1:8000/api/v1/my_courses/student/

GET http://127.0.0.1:8000/api/v1/my_courses/student/

Params Authorization Headers (9) Body Pre-request Script Tests Settings

Key	Value	Description
Accept-Encoding	gzip, deflate, br	
Connection	keep-alive	
Authorization	Bearer eyJhbGciOiJIUzI1NilsInR5cCI6IkpXVCJ9eyJ0b2...	

Body Cookies Headers (10) Test Results Status: 200

Pretty Raw Preview Visualize JSON

```
1  {
2      "courses": [
3          {
4              "id": 1,
5              "title": "Python Programming Language",
6              "description": "Python is a high-level, general-purpose programming language.",
7              "teacher": {
8                  "first_name": "Saule",
9                  "last_name": "Apai",
10                 "email": "teacher1@gmail.com"
11             }
12         }
13     ]
14 }
```

СПИСОК КУРСОВ STUDENT-A

PUT /dashboard/courses/{course_id}/lessons/{lesson_id}/change/

GET /dashboard/courses/{course_id}/lessons/{lesson_id}/comments/

POST /dashboard/courses/{course_id}/lessons/{lesson_id}/comments/

COMMENTS

```
1  from django.db import models
2  from accounts.models import User
3  from courses.models import Lesson
4
5
5 usages  ↗ Azamat
6  class Comment(models.Model):
7      lesson = models.ForeignKey(Lesson, on_delete=models.CASCADE, related_name='comments')
8      user = models.ForeignKey(User, on_delete=models.CASCADE)
9      parent = models.ForeignKey('self', on_delete=models.CASCADE, null=True, blank=True, related_name='replies')
10     content = models.TextField()
11     created_at = models.DateTimeField(auto_now_add=True)
12     updated_at = models.DateTimeField(auto_now=True)
13
14  ↗ def __str__(self):
15      ↗ return f"Comment by {self.user} on Lesson {self.lesson}"
16
```

ЧЕГО ПРИДЕРЖИВАЛСЯ?

CODING PRINCIPLES

YAGNI

Do the NEEDED



KISS

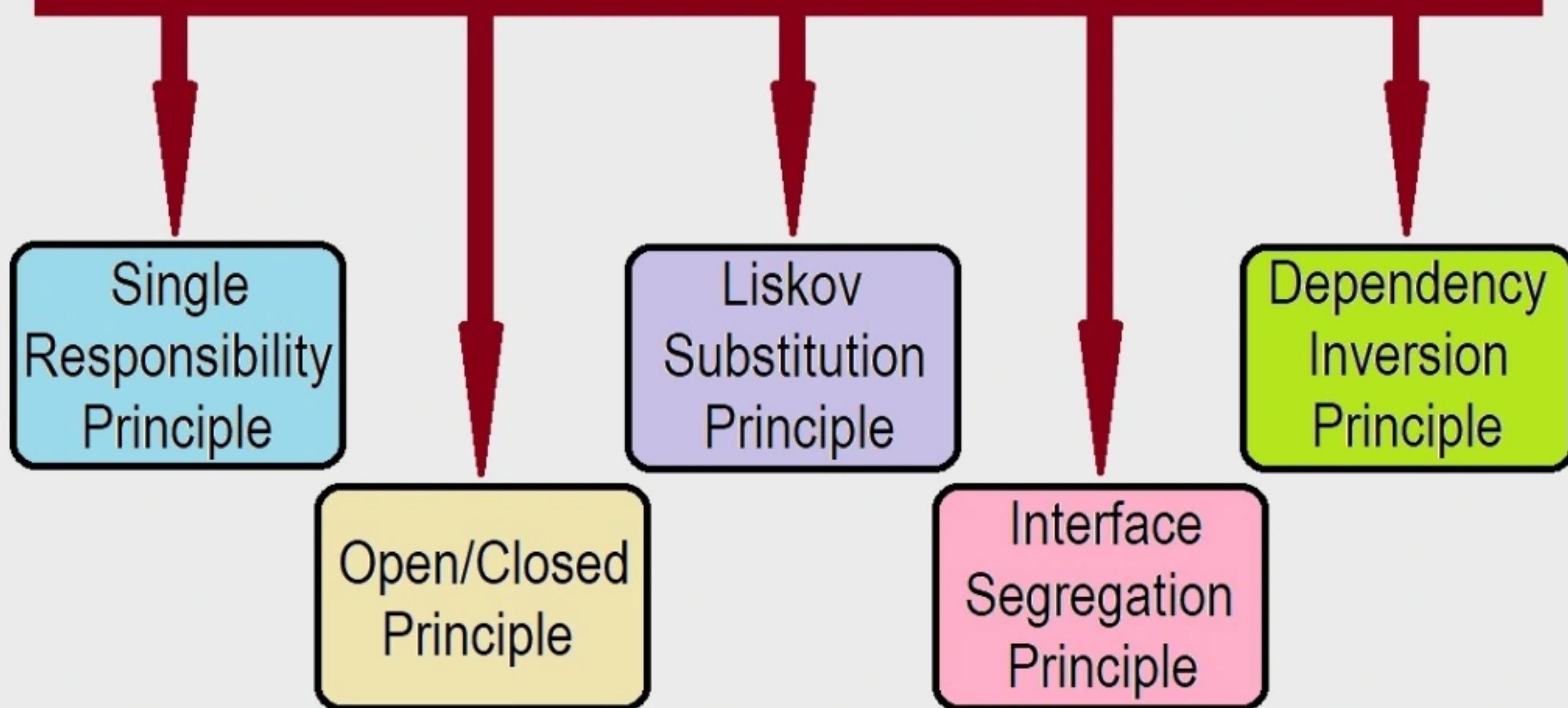
Do it SIMPLE



DRY

Do it ONCE



A screenshot of a file explorer window showing a project structure for a Python application. The structure includes:

- courses (directory)
 - migrations (directory)
 - routes (directory)
 - course (directory)
 - __init__.py
 - changeCourse.py
 - createCourse.py
 - deleteCourse.py
 - getAllCourse.py
 - getCourseById.py
 - joinToCourse.py
 - lesson (directory)
 - __init__.py
 - changeLesson.py
 - createLesson.py
 - deleteLesson.py
 - getLessonById.py
 - getLessonsByCourseId.py
 - urls (directory)
 - __init__.py
 - v1.py
 - __init__.py
 - serializers (directory)
 - __init__.py
 - course.py
 - lesson.py
 - __init__.py
 - admin.py
 - apps.py
 - models.py
 - tests.py
 - views.py

ТАКЖЕ ИСПОЛЬЗОВАЛ

permissions
(права для разных
типов пользователей)

decouple
(для получения
данных из .env)

serializers

send_mail

customUser

.gitignore

Docker Desktop Update to latest

Q Search for local and remote images, containers, and more...

Ctrl+K



Sign in



Containers

Images

Volumes

Dev Environments BETA

Learning Center

Extensions

Add Extensions

< demodayproject

C:\Users\AZAMAT\PycharmProjects\DemodayProject



demodayproject-db-1

postgres



Running

demodayproject-djan...

demodayproject-django



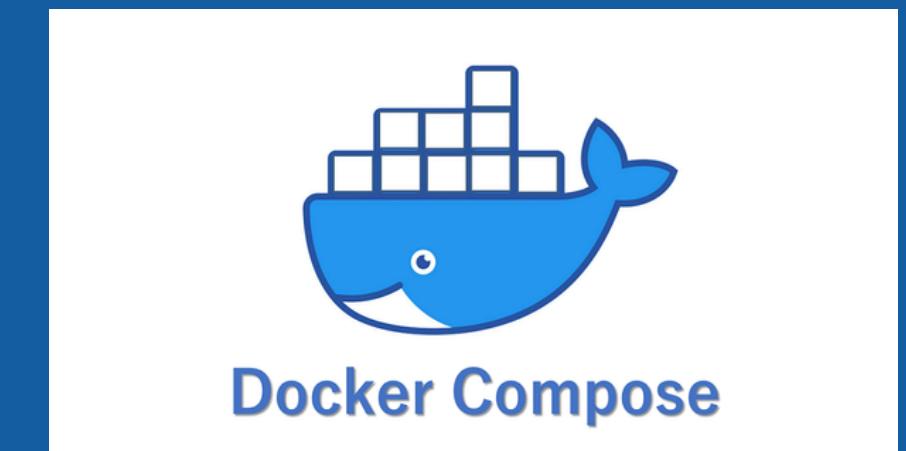
Running

8000:8000

```
ing on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
2023-07-15 00:38:27 demodayproject-db-1 | 2023-07-14 21:38:27.9
ase system was shut down at 2023-07-13 18:19:14 UTC
2023-07-15 00:38:27 demodayproject-db-1 | 2023-07-14 21:38:27.984 UTC [1] LOG:  database
se system is ready to accept connections
2023-07-15 00:43:28 demodayproject-db-1 | 2023-07-14 21:43:28.071 UTC [27] LOG:  check
point starting: time
2023-07-15 00:43:28 demodayproject-db-1 | 2023-07-14 21:43:28.190 UTC [27] LOG:  check
point complete: wrote 3 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0
.056 s, sync=0.008 s, total=0.120 s; sync files=2, longest=0.005 s, average=0.004 s; distan
ce=0 kB, estimate=0 kB
2023-07-15 00:48:28 demodayproject-db-1 | 2023-07-14 21:48:28.781 UTC [27] LOG:  check
point starting: time
2023-07-15 00:48:30 demodayproject-db-1 | 2023-07-14 21:48:30.048 UTC [27] LOG:  check
point complete: wrote 13 buffers (0.1%); 0 WAL file(s) added, 0 removed, 0 recycled; write=
1.221 s, sync=0.014 s, total=1.272 s; sync files=13, longest=0.007 s, average=0.002 s; dist
ance=12 kB, estimate=12 kB
2023-07-15 01:33:28 demodayproject-db-1 | 2023-07-14 22:33:28.645 UTC [27] LOG:  check
point starting: time
2023-07-15 01:33:28 demodayproject-db-1 | 2023-07-14 22:33:28.816 UTC [27] LOG:  check
point complete: wrote 2 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=
0.116 s, sync=0.009 s, total=0.187 s; sync files=2, longest=0.006 s, average=0.005 s; distan
ce=7 kB, estimate=11 kB
2023-07-15 01:38:28 demodayproject-db-1 | 2023-07-14 22:38:28.856 UTC [27] LOG:  check
point starting: time
2023-07-15 01:38:29 demodayproject-db-1 | 2023-07-14 22:38:29.727 UTC [27] LOG:  check
point complete: wrote 9 buffers (0.1%); 0 WAL file(s) added, 0 removed, 0 recycled; write=
0.814 s, sync=0.018 s, total=0.871 s; sync files=9, longest=0.009 s, average=0.002 s; distan
ce=9 kB, estimate=11 kB
```

```
docker-compose.yml × Dockerfile ×
1 ► FROM python:3.11
2
3 ENV PYTHONUNBUFFERED 1
4
5 WORKDIR /code
6
7 COPY requirements.txt .
8
9 RUN pip install -r requirements.txt
10
11 COPY . .
12
13 EXPOSE 8000
14
15 CMD ["python", "manage.py", "runserver"]
16
```

```
docker-compose.yml × Dockerfile ×
1 version: '3'
2
3 services:
4   db:
5     image: postgres
6     environment:
7       POSTGRES_DB: onlineschoolwebsite
8       POSTGRES_USER: postgres
9       POSTGRES_PASSWORD: aza2003
10  django:
11    build:
12      context: .
13      dockerfile: Dockerfile
14    command: bash -c "python manage.py makemigrations && python manage.py migrate && python manage.py runserver 0.0.0.0:8000"
15    volumes:
16      - ./code
17    ports:
18      - "8000:8000"
19    depends_on:
20      - db
```





PostgreSQL

PYTHON DOTENV

MANAGE ENVIRONMENT VARIABLES

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.postgresql_psycopg2',  
        'NAME': config('NAME'),  
        'USER': 'postgres',  
        'PASSWORD': config('PASSWORD'),  
        'HOST': 'db',  
        'PORT': '5432',  
    },  
}
```

A screenshot of a code editor showing a file named '.env'. The file contains the following environment variables:

Variable	Value
EMAIL_HOST_USER	onlajnskola95
EMAIL_HOST_PASSWORD	jowobiizr
SECRET_KEY	django-insecure-z8
PASSWORD	aza2003
NAME	onlineschoolwebsite

Спасибо за внимание



azambylbaev@gmail.com

