```
pip install -r requirements.txt

#1. Person:

            from django.db import models


            class Person(models.Model):
                  name = models.CharField(
                        max_length=30,
                  )
                  age = models.PositiveIntegerField()


#2. Blog:

            from django.db import models


            class Blog(models.Model):
                  post = models.TextField()
                  author = models.CharField(
                        max_length=35,
                  )

#3. Weather Forecast:

            class WeatherForecast(models.Model):
                  date = models.DateField()
                  temperature = models.FloatField()
                  humidity = models.FloatField()
                  precipitation = models.FloatField()


#4. Recipe:

            class Recipe(models.Model):
                  name = models.CharField(
                        max_length=100,
                        unique=True,
                  )
                  description = models.TextField()
                  ingredients = models.TextField()
                  cook_time = models.PositiveIntegerField()
                  created_at = models.DateTimeField(
                        auto_now_add=True,
                  )

#5. Product:

            class Product(models.Model):
                  name = models.CharField(
                        max_length=70,
                  )
                  description = models.TextField()
                  price = models.DecimalField(
                        max_digits=10,
                        decimal_places=2,
                  )
```

```
                    created_at = models.DateTimeField(
                          auto_now_add=True,
                    )
```

#6. User Profile:

```
            class UserProfile(models.Model):
                    username = models.CharField(
                          max_length=65,
                          unique=True
                    )
                    first_name = models.CharField(
                          max_length=40,
                          unique=True
                    )
                    last_name = models.CharField(
                          max_length=40,
                          unique=True,
                    )
                    email = models.EmailField(
                          unique=True,
                          default="students@softuni.bg"
                    )
                    bio = models.TextField(
                          max_length=120,
                    )
                    profile_image_url = models.URLField()
                    created_at = models.DateTimeField(
                          auto_now_add=True,
                    )
```

#7. Exercise:

```
            class Exercise(models.Model):
                    name = models.CharField(
                          max_length=100,
                    )
                    description = models.TextField()
                    difficulty_level = models.CharField(
                          max_length=20,
                    )
                    duration_minutes = models.PositiveIntegerField()
                    equipment = models.CharField(
                          max_length=90,
                    )
                    video_url = models.URLField(
                          null=True,
                          blank=True,
                    )
                    calories_burned = models.PositiveIntegerField(
                          default=0,
                    )
                    is_favourite = models.BooleanField(
                          default=False,
                    )
```

#8. Book:

```
            class Book(models.Model):
                  GENRE_CHOICES = [
                        # in real world can use ("F", "Fiction") in order to save
memory space
                        ("Fiction", "Fiction"),
                        ("Non-Fiction", "Non-Fiction"),
                        ("Science Fiction", "Science Fiction"),
                        ("Horror", "Horror")
                  ]

                  title = models.CharField(
                        max_length=30,
                  )
                  author = models.CharField(
                        max_length=100,
                  )
                  genre = models.CharField(
                        max_length=20,
                        choices=GENRE_CHOICES
                  )
                  publication_date = models.DateField(
                        editable=False,
                        auto_now_add=True,
                  )
                  price = models.DecimalField(
                        max_digits=8,
                        decimal_places=2
                  )
                  is_available = models.BooleanField(
                        default=True,
                  )
                  rating = models.FloatField()
                  description = models.TextField()

                  def __str__(self):
                        return {self.title}


            8.1 Register in Django Admin
            @admin.register(Book)
            class BookAdmin(admin.ModelAdmin):
                  pass


            ### ADDITIONAL EXERCISE FOR VALIDATORS ###

            #1. Create new file validators.py:
                  def validate_age(value):
                        if value <= 0 or value > 120:
                              raise ValidationError('Age must be between 0 and
120')

            #2. In models.py:
            class CustomPersonExample(models.Model):
                  name=models.CharField(
                        max_length=30,
                  )
                  age = models.PositiveIntegerField(
                        validators=[
                              validate_age <-- or use built-in validator:
```

```
MaxValueValidator(120)
                                    ]
                            )
```