К.О.Н.С.П.Е.К.Т

Предупреждаю: мы не будем писать, то что мы давно учим, поэтому начнем с того что недавно учим

models.py/Player:

**has\_the\_golden\_ball = models.BooleanField()**

terminal:

**python manage.py makemigrations**

terminal:

**python manage.py migrate**

terminal:

**python manage.py shell**

terminal-manage:

**from statsapp.models import Player**

terminal-manage:

**Player.objects.create(name=”Lionel Messi”,goals = 4, has\_the\_golden\_ball = True)**

**Player.objects.create(name=”Cristiano Ronaldo”,goals = 1, has\_the\_golden\_ball =True)**

**Player.objects.create(name=”Kylian Mbappe”,goals = 5, has\_the\_golden\_ball = True)**

**Player.objects.create(name=”Marcus Rashford”,goals = 3, has\_the\_golden\_ball = False)**

**Player.objects.create(name=”Olivion Giroud”,goals = 2, has\_the\_golden\_ball = False)**

templates/statsapp/listplayers.html:

**!***+TAB or +Enter*

templates/statsapp/player.html:

**!***+TAB or +Enter*

terminal-manage:

**Player.objects.order\_by(“goals”):**

**Players.objects.order\_by(“-goals”):**

views:

from statsapp.models import Player

**def top\_3\_players(request):**

**list\_players = Player.objects.order\_by(“-goals”)**

**list\_players = list\_players[0:3]**

**return render(request,”statsapp/listplayers.html”,context={“title”:”Топ 3 гравці за голами”,”list\_players”:list\_players}**

listplayers.html/title:

**{{title}}**

listplayers.html/body:

**{% for player in list\_players %}**

**<h2>{{player.name}}</h2>**

**<p>Кількість голів на Чемпіонаті Світі:{{player.goals}}</p>**

**<hr>**

**{% endfor %}**urls:

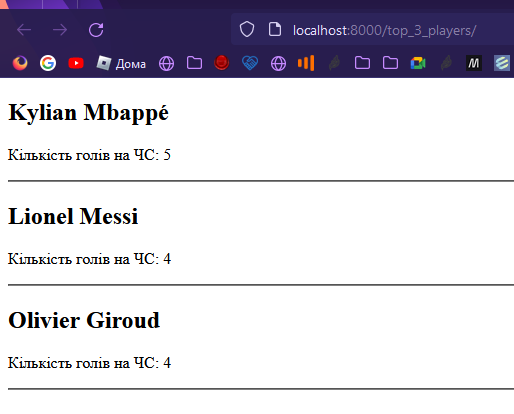
**from statsapp.views import \***

urls/urlpatterns:

**path(‘top\_3\_players/’,top\_3\_players),**

terminal:

**python manage.py runserver**

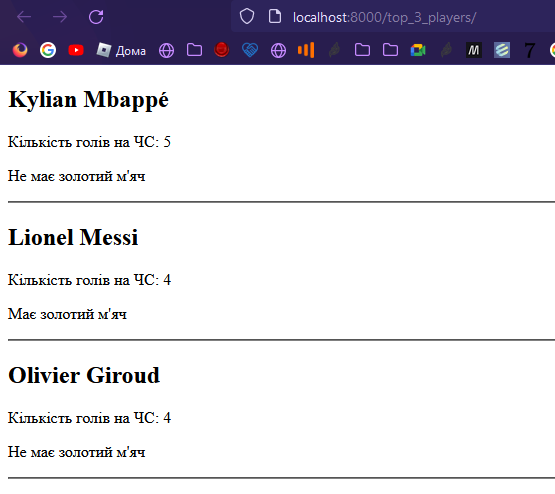
listplayers.html/body/for after tag “p”:

**{% if player.has\_the\_golden\_ball %}**

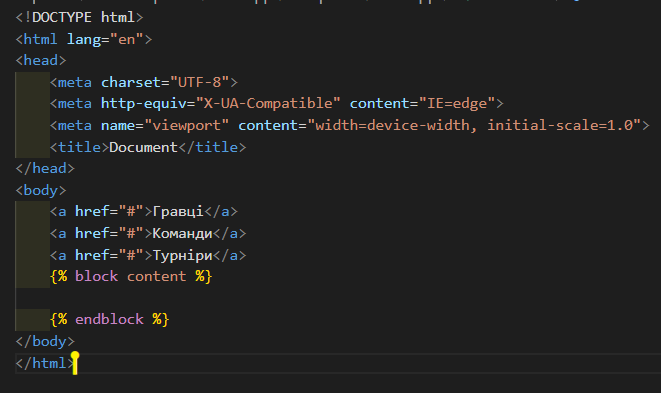
**<p>Має золотий м’яч</p>**

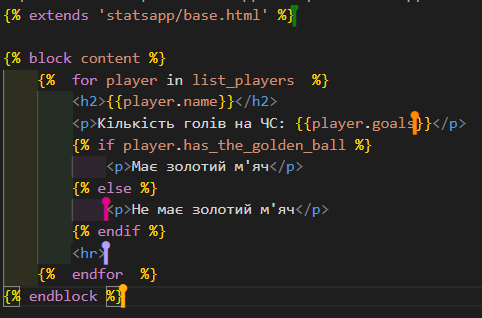
**{% else %}**

**<p>Не має золотий м’яч</p>**

**{% endif %}**

templates/statsapp/base.html/body:

listplayers.html/*remove code besides*:

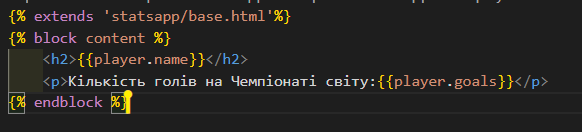
views.py:

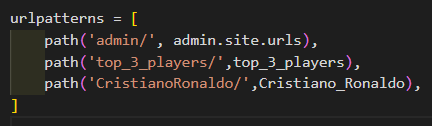
**def Cristiano\_Ronaldo(request):**

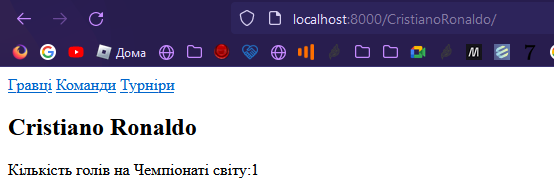
**player = Player.objects.get(pk=2)**

**return render(request,”stats.app/player.html”,context={“title”:”Cristiano Ronaldo”,”player”:player})**

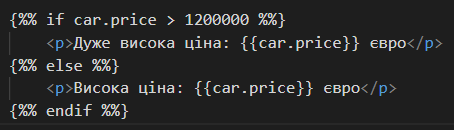
player.html:

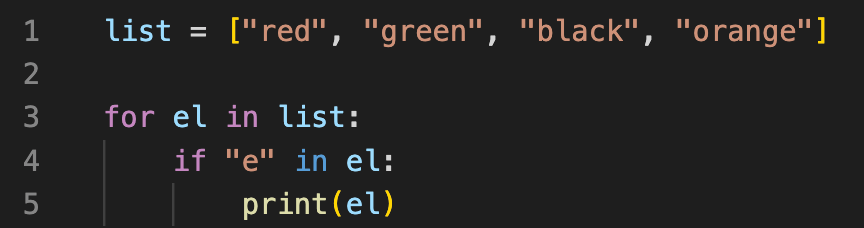
urls.py:



ТЕСТ

1.Через який параметр функції render ми передаємо дані, що хочемо викорисовувати у шаблоні?: **context**

2. Вкажіть синтаксично не правильно створену умову за допомогою шаблонизатора Jinja (на картинках показано всю умову):

3.

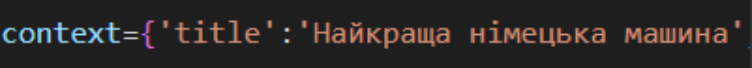
:**red**

**green**

**orange**

4. Який метод сортує записи БД за певним полем?: **order\_by**

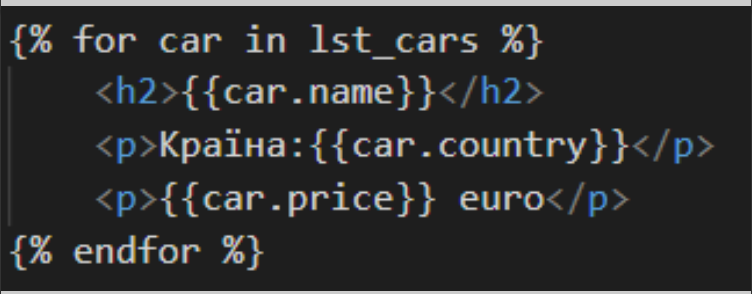
5. Передаємо шаблон параметр title чурез context. Як відобразити цей параметр у шаблоні?

:**{{ title }}**

6. Який тег використовуємо для підключення базового шаблона у html-документ?: **{% extends '...' %}**

7. Вкажіть назву типу даних кортеж?: **tuple**

8.Вкажіть правильно створений цикл за допомогою шаблонизатора Jinja (на картинках показано весь цикл) :

9.

: **('list', ['red', 'green'])**

**('color', 'red')**

10. Чи обов'язково при створенні циклу за допомогою Jinja писати {% endfor %} у кінці тіла циклу? : **True**