

# Аутентификация и авторизация

Аутентификация –  
процесс подтверждения  
личности пользователя

Авторизация – процесс  
подтверждения  
обладанием  
определёнными правами  
и возможностями

```
urlpatterns += [ path('accounts/',include('django.contrib.auth.urls')),  
]
```

#тем самым мы добавляем набор паттернов для управления аутентификацией, помимо этого следует реализовать новые шаблоны для входа, выхода и изменения пароля

## urls.py

Настройки маршрутизации

```
class Post(models.Model):  
    class Meta:  
        permissions = (("can_post", "Can post"), ("can_edit",  
"Can edit"))
```

## Permissions

операции, которые могут быть  
выполнены над моделью  
определёнными  
пользователями

```
#template:
```

```
{% if perms.blog.can_edit %}
```

```
    <!-- show edit button -->
```

```
{% endif %}
```

```
#view:
```

```
from django.contrib.auth.decorators import  
permission_required
```

```
@permission_required('blog.can_edit')
```

```
def edit(request):
```

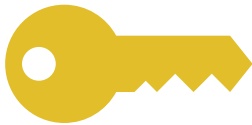
```
#some logic
```

## Проверка разрешений

Может выполняться как на  
представлении, так и в шаблоне



# Сессии – механизм, используемый в сети Интернет, для имитации «состояния» взаимодействия между сайтом и определённым браузером



Сессии позволяют хранить данные по принципу ключ-значение



Django использует cookie в котором хранится идентификатор сессии, все данные сессии при этом хранятся в базе данных



Можно изменить способ хранения данных сессии на другой – кэш, «безопасный» cookie, файл, etc

```
INSTALLED_APPS = [  
    ...  
    'django.contrib.sessions',  
    ....  
]  
MIDDLEWARE = [  
    ...  
    'django.contrib.sessions.middleware.SessionMiddleware',  
    ....  
]
```

## Сессии

Включение механизма сессий —  
settings.py

```
latest = request.session['latest_post'] #получаем значение из сессии
```

```
latest = request.session.get('latest_post', 'There are no new posts at the moment') #получаем значение из сессии либо значение по-умолчанию
```

```
request.session['latest_post'] = 'New post' #устанавливаем значение
```

```
del request.session['latest_post'] #удаляем значение из сессии
```

```
request.session.modified = True #указываем в явном виде, что объект сессии был изменён
```

## Сессии

Работа с сессией