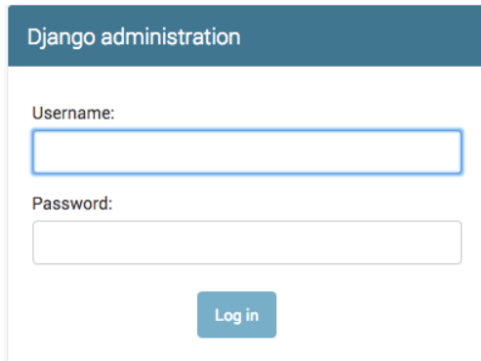


How to change 'Django administration' text

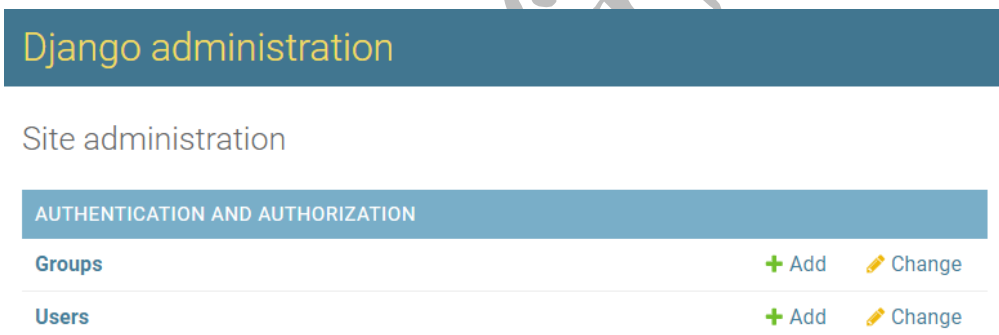
By default Django admin shows 'Django administration'.

The text is at these pages:

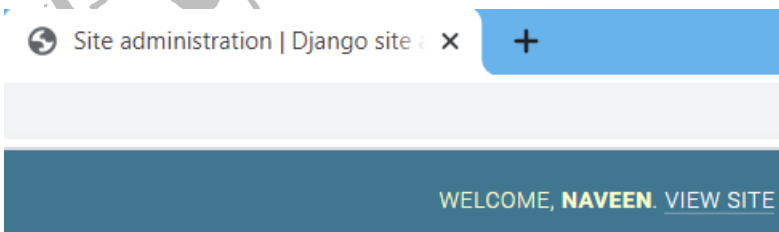
- Login Page



- The listview page



- The HTML title tag



We can make the three changes in **urls.py**:

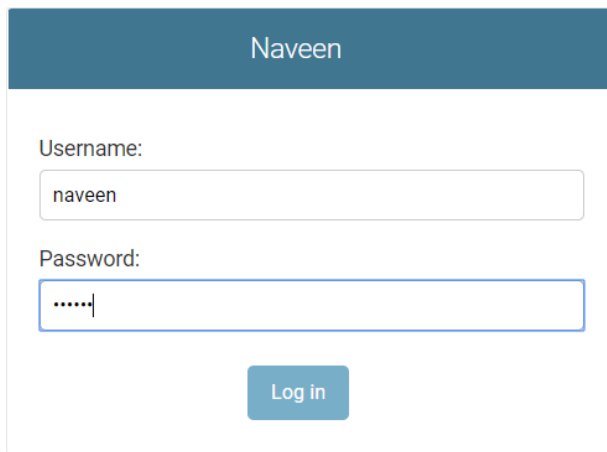
```
admin.site.site_header = "Naveen"
```

```
admin.site.site_title = "Python With Naveen"
```

```
admin.site.index_title = "Welcome to Python With Naveen Page"
```

After changing

- Login Page



Naveen

Username:

Password:

Log in

- The listview page

Naveen

Welcome to Python With Naveen Page

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add

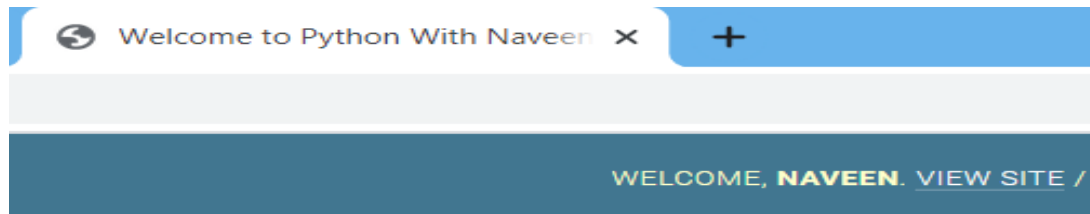
Change

Users

+ Add

Change

- The HTML title tag



How to set the plural text for a model

1) Default

models.py

```
from django.db import models
```

```
class ProductModel(models.Model):
```

```
    number = models.AutoField(primary_key = True)
```

```
    name = models.CharField(max_length=100,unique=True)
```

```
    price = models.FloatField()
```

admin.py

```
from django.contrib import admin
```

```
from app1.models import ProductModel
```

```
admin.site.register(ProductModel)
```

Output in admin window

Welcome to Python With Naveen Page



2) If we use verbose_name = "Product"

models.py

```
from django.db import models
```

```
class ProductModel(models.Model):
```

```
    number = models.AutoField(primary_key = True)
```

```
    name = models.CharField(max_length=100,unique=True)
```

```
    price = models.FloatField()
```

```
    class Meta:
```

```
        verbose_name = "Product"
```

Output in admin window

Welcome to Python With Naveen Page

APP1

Products

+ Add

Change

By default admin will show the name of your model appended with an "s".

3) If we use verbose_name_plural = "Product"

models.py

```
from django.db import models
```

```
class ProductModel(models.Model):
```

```
    number = models.AutoField(primary_key = True)
```

```
    name = models.CharField(max_length=100,unique=True)
```

```
    price = models.FloatField()
```

```
    class Meta:
```

```
        verbose_name_plural = "Product"
```

Output in admin window

Welcome to Python With Naveen Page

APP1

Product

+ Add

✎ Change

How to create two independent admin sites

1) Default admin

models.py

```
from django.db import models
```

```
class EmployeeModel(models.Model):
```

```
    number = models.AutoField(primary_key=True)
```

```
    name = models.CharField(max_length=100)
```

```
    salary = models.FloatField()
```

```
    class Meta:
```

```
        verbose_name_plural = "Employee"
```

admin.py

```
admin.site.register(EmployeeModel)
```

2) Extra admin

models.py

```
from django.db import models
```

```
class ProductModel(models.Model):
```

```
    number = models.AutoField(primary_key=True)
```

```
    name = models.CharField(max_length=100, unique=True)
```

Youtube : <https://www.youtube.com/c/pythonwithnaveen>

Facebook : <https://www.facebook.com/groups/pythonwithnaveen/>

```
price = models.FloatField()
```

```
class Meta:
```

```
    verbose_name_plural = "Product"
```

```
admin.py
```

```
class KumarAdminSite(AdminSite):
```

```
    site_header = "Naveen Kumar"
```

```
    site_title = "Python With Naveen Kumar"
```

```
    index_title = "Welcome to Python With Naveen Kumar Page"
```

```
kumar_admin_site = KumarAdminSite(name='naveen kumar')
```

```
kumar_admin_site.register(ProductModel)
```

```
urls.py
```

```
path('naveen_admin/', admin.site.urls),
```

```
path('kumar_admin/', kumar_admin_site.urls),
```

Caution Multiple Django admin sites don't stop user

permissions, authentication or url visibility. By default, all Django users can use the same credentials and are assigned the same permissions across multiple Django admin sites. This means it's up to you to enforce effective model permissions across all Django admin sites, limit model registration for each Django admin site, as well as avoid easy to guess Django admin urls to avoid unintended access.

How to remove default apps from Django admin

Django will include `django.contrib.auth` in `INSTALLED_APPS`, which means *User* and *Groups* models are included in admin automatically.

Welcome to Python With Naveen Page

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add

Change

Users

+ Add

Change

If you want to remove it, you will have to unregister them.

```
from django.contrib.auth.models import User, Group
```

```
admin.site.unregister(User)
```

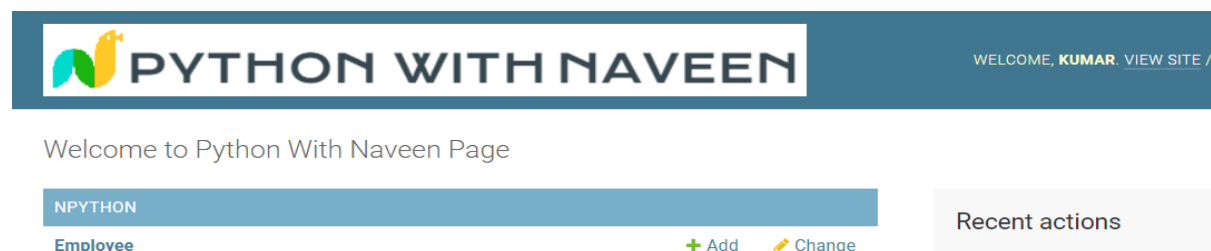
```
admin.site.unregister(Group)
```

How to add a logo to Django admin?

Copy the `base_site.html` from the admin app to `templates\admin` folder. Replace there default text in *branding* block with:

```
<a href="{% url 'admin:index' %}">  
    
</a>
```

Output

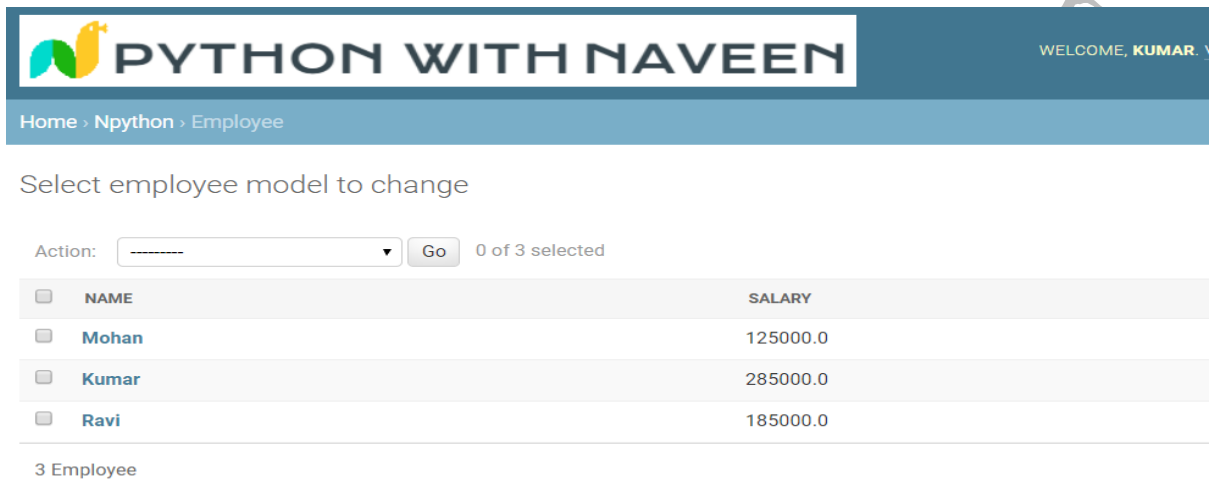


The screenshot shows the Django Admin interface. At the top, there is a blue header bar with the Python With Naveen logo on the left and the text "WELCOME, KUMAR. VIEW SITE /" on the right. Below the header, the text "Welcome to Python With Naveen Page" is displayed. Underneath, there is a blue bar with the text "NPYTHON" and "Employee". To the right of this bar, there are two links: "+ Add" and "Change". Further right, there is a box labeled "Recent actions".

admin **list display** example

```
@admin.register(EmployeeModel)
class EmployeeAdmin(admin.ModelAdmin):
    list_display = ('name', 'salary')
```

Output



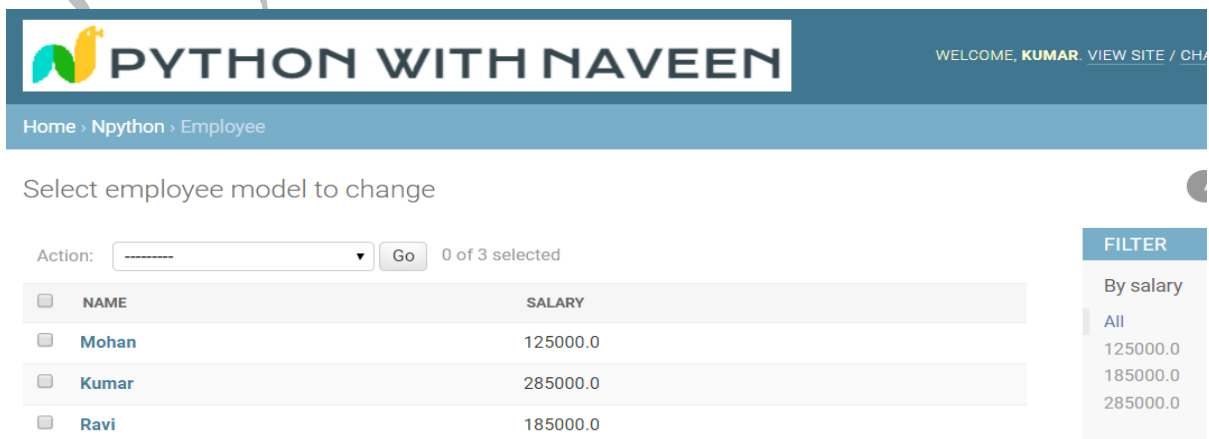
NAME	SALARY
Mohan	125000.0
Kumar	285000.0
Ravi	185000.0

3 Employee

admin **list filter** example

```
@admin.register(EmployeeModel)
class EmployeeAdmin(admin.ModelAdmin):
    list_display = ('name', 'salary')
    list_filter = ('salary',)
```

Output



NAME	SALARY
Mohan	125000.0
Kumar	285000.0
Ravi	185000.0

3 Employee

FILTER

By salary

☒ All

☐ 125000.0

☐ 185000.0

☐ 285000.0

How to add date based filtering in Django admin

You can add a date based filtering on any date field by setting the `date_hierarchy`.

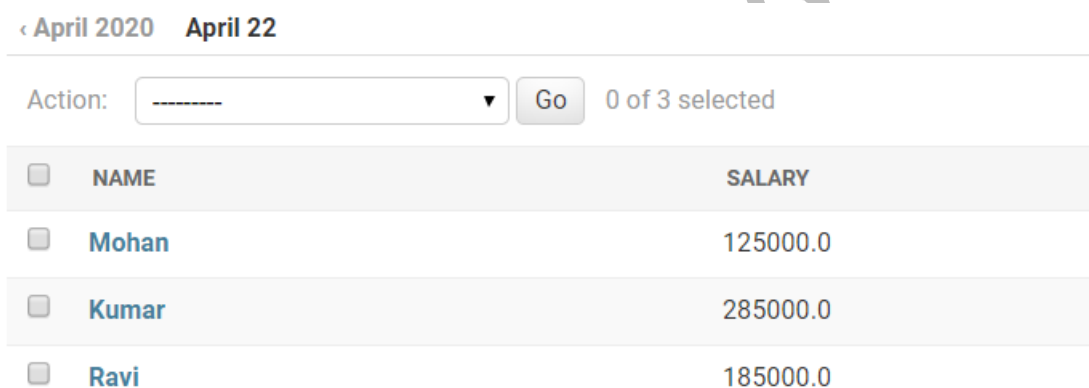
```
@admin.register(EmployeeModel)
class EmployeeAdmin(admin.ModelAdmin):
```

```
.....
```

```
.....
```

```
    date_hierarchy = 'jdate'
```

Output



◀ April 2020 April 22


Action: Go 0 of 3 selected

<input type="checkbox"/>	NAME	SALARY
<input type="checkbox"/>	Mohan	125000.0
<input type="checkbox"/>	Kumar	285000.0
<input type="checkbox"/>	Ravi	185000.0

How to mark a field as readonly in admin

```
@admin.register(EmployeeModel)
class EmployeeAdmin(admin.ModelAdmin):
    list_display = ('name', 'salary', 'photo')
    readonly_fields = ['name']
```

Output

 PYTHON WITH NAVEEN

Home › Npython › Employee › Mohan

Change employee model

Salary:

125000.0

Photo:

Currently: employee_images/mohan.jpg
Change: No file chosen

Name:

Mohan