

[Understand]

on_delete in Django Models



Introduction to on_delete in model

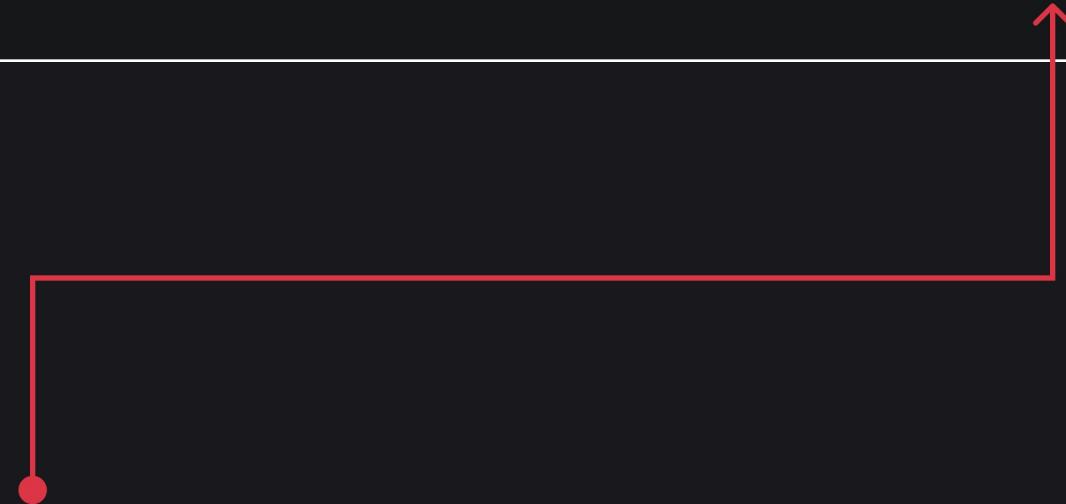
In django models, on_delete is an attribute, used with OneToOneField or ForeignKey fields. It specify the behaviour of child related object, if parent model referenced object is deleted.

for eg., let's say we have a Author model and Book model and this have foreign key relation with Author model

```
from django.db import models

class Author(models.Model):
    name = models.CharField(max_length=100)

class Book(models.Model):
    title = models.CharField(max_length=100)
    author = models.ForeignKey(Author, on_delete=models.CASCADE)
```



Here we have specified the `on_delete` attribute and we have set to `CASCADE`, it means when parent referenced object is delete, then it will also delete the child referenced object

on_delete options

Here are the available options for the on_delete attribute:

1. **CASCADE**: When the referenced object is deleted, it also delete the objects that have an OneToOneField or ForeignKey referencing it.

This effectively cascades the delete operation.



```
author = models.ForeignKey(Author, on_delete=models.CASCADE)
```

2. **PROTECT**: Prevent deletion of the referenced object by raising a ProtectedError exception.



```
author = models.ForeignKey(Author, on_delete=models.PROTECT)
```

3. **SET_NULL**: Set the ForeignKey or OneToOneField to NULL when the referenced or parent object is deleted. And make sure null is True.



```
author = models.ForeignKey(Author, on_delete=models.SET_NULL, null=True)
```

4. **SET_DEFAULT**: Set the ForeignKey or OneToOneField to its default value when the referenced object is deleted.



```
author = models.ForeignKey(Author,  
                           on_delete=models.SET_DEFAULT, default=None)
```

5. **DO NOTHING**: Do nothing when the referenced object is deleted.



```
author = models.ForeignKey(Author, on_delete=models.DO_NOTHING)
```

6. RESTRICT: Restrict deletion of the referenced object by raises a RestrictedError exception.



```
author = models.ForeignKey(Author, on_delete=models.RESTRICT)
```

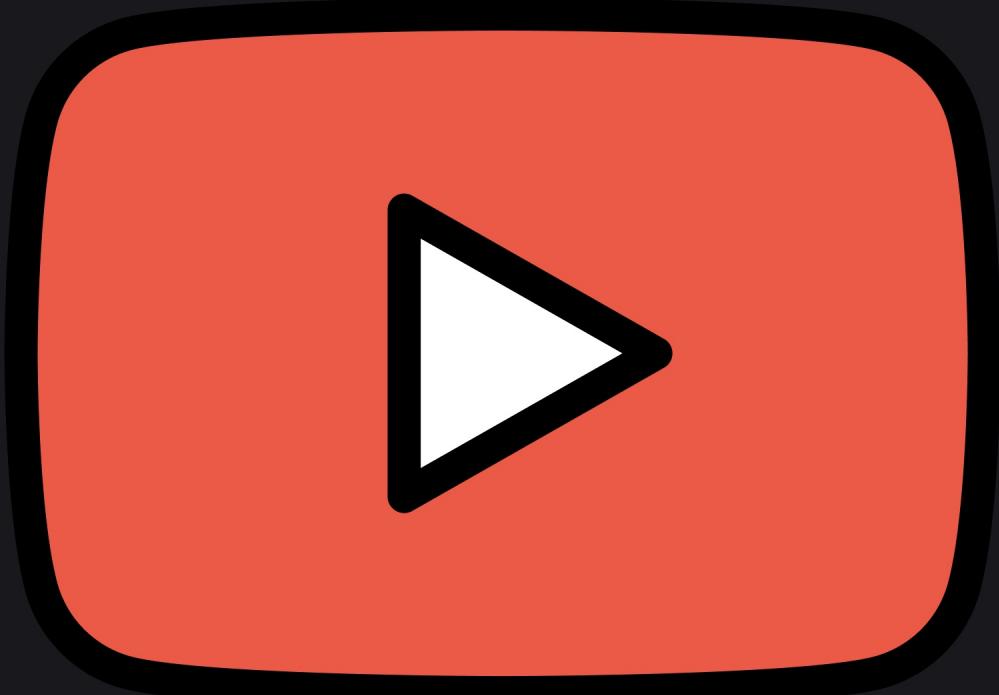
Conclusion

on_delete is a required attribute when we working with OneToOneField or ForeignKey fields, we can use any type of on_delete options based on our models requirements

For more django
related content

Subscribe

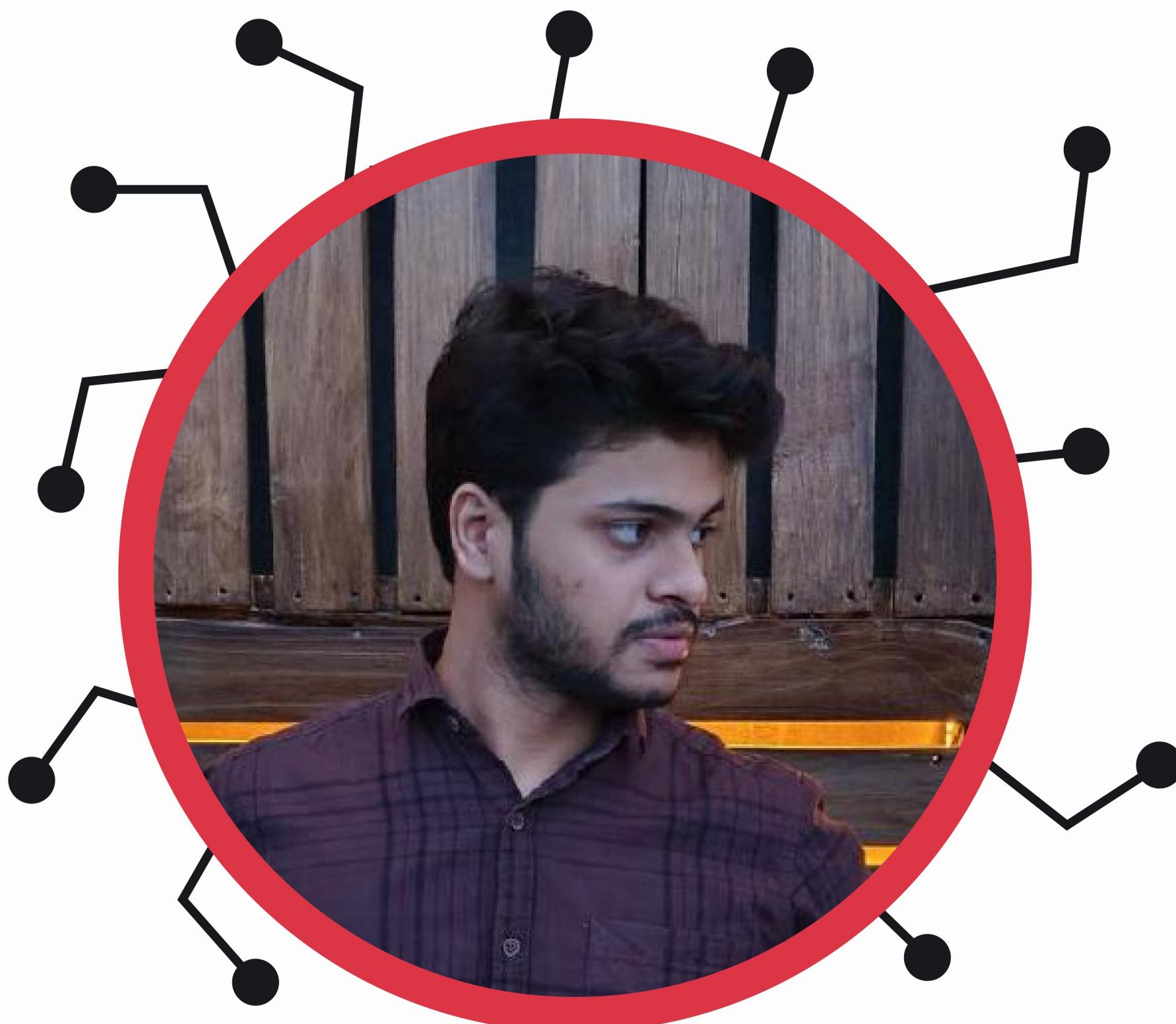
To My Youtube
Channel



Link in bio

DID YOU FIND THIS HELPFUL

Let me know in the comment



Aashish Kumar

Software Engineer

Follow for more ❤