

ModelSerializer Class

The ModelSerializer class provides a shortcut that lets you automatically create a Serializer class with fields that correspond to the Model fields.

The ModelSerializer class is the same as a regular Serializer class, except that:

- It will automatically generate a set of fields for you, based on the model.
- It will automatically generate validators for the serializer, such as `unique_together` validators.
- It includes simple default implementations of `create()` and `update()`.

Create ModelSerializer Class

- Create a separate serializers.py file to write all serializers

```
from rest_framework import serializers
```

```
class StudentSerializer(serializers.ModelSerializer):
```

```
    class Meta:
```

```
        model = Student
```

```
        fields = ['id', 'name', 'roll', 'city']
```

- `fields = '__all__'`
- `exclude = ['roll']`

Create ModelSerializer Class

```
from rest_framework import serializers

class StudentSerializer(serializers.ModelSerializer):
    name = serializers.CharField(read_only=True)

    class Meta:
        model = Student
        fields = ['id', 'name', 'roll', 'city']

class StudentSerializer(serializers.ModelSerializer):
    class Meta:
        model = Student
        fields = ['id', 'name', 'roll', 'city']
        read_only_fields = ['name', 'roll']
```

Create ModelSerializer Class

```
from rest_framework import serializers

class StudentSerializer(serializers.ModelSerializer):

    class Meta:
        model = Student
        fields = ['id', 'name', 'roll', 'city']
        extra_kwargs = {'name': {'read_only': True}}
```

ModelSerializer Validation

```
from rest_framework import serializers

class StudentSerializer(serializers.ModelSerializer):

    class Meta:
        model = Student
        fields = ['id', 'name', 'roll', 'city']

    def validate_roll(self, value):
        if value >= 200 :
            raise serializers.ValidationError('Seat Full')
        return value
```