

PoliceStation

PoliceStaff class

CODE

```
class PoliceStaff
{
    private string policeID;
    private string policeName;
    private string policeDesignation;
    private ulong policeContactNumber;
    private string policeBranch;

    public void RecruitPoliceOfficer()
    {
        //todo
    }

    public void EndOfService()
    {
        //todo
    }

    public void DisplayPoliceOfficer()
    {
        //todo
    }

    public void JobProfile()
    {
        //todo
    }

    public void Documentation()
    {
        //todo
    }
}
```

UML diagram

PoliceStaff

- policeID : string
- policeName : string
- policeDesignation : string
- policeContactNumber : ulong
- policeBranch : string

+ RecruitPoliceOfficer() : void
+ EndOfService() : void
+ DisplayPoliceOfficer() : void
+ JobProfile() : void
+ Documentation() : void

PoliceStation

Complaints class

Code

```
class Complaints
{
    private string complaintID;
    private string complainantName;
    private string complaint;
    private ulong complainantConatactNumber;
    private string complainantAddress;

    public void FileComplaint()
    {
        //todo
    }

    public void ResolveComplaint()
    {
        //todo
    }

    public void UpdateComplaint()
    {
        //todo
    }

    public void DisplayComplaint()
    {
        //todo
    }
}
```

UML Diagram

Compliant class

- complaintID : string
- complainantName : string
- complaint : string
- complainantConatactNumber : string
- complainantAddress : string

+ FileComplaint() : void
+ ResolveComplaint() : void
+ UpdateComplaint() : void
+ DisplayComplaint() : void

PoliceStation

Convict class

Code

```
class Convict
{
    private string convictID;
    private string convictName;
    private string convictCrime;
    private string convictVerdict;
    private byte convictAge;

    public void AddConvict()
    {
        //todo
    }

    public void DeleteConvict()
    {
        //todo
    }

    public void CaseProceedings()
    {
        //todo
    }

    public void DisplayConvict()
    {
        //todo
    }

    public void Documentation()
    {
        //todo
    }
}
```

UML Diagram

Convict class

```
- convictID : string
- convictName : string
- convictCrime : string
- convictVerdict : string
- convictAge : byte

+ AddConvict() : void
+ DeleteConvict() : void
+ CaseProceedings() : void
+ DisplayConvict() : void
+ Documentation() : void
```

PoliceStation

Administration class

Code

```
class Administration
{
    private uint salaries;
    private string policeStationAddress;
    private byte equipmentPerStaion;
    private string equipmentType;
    private byte staffPerStation;

    public void AdministrationOfPoliceStation()
    {
        //todo
    }

    public void CoordinatePoliceStations()
    {
        //todo
    }

    public void Transactions()
    {
        //todo
    }

    public void Documentation()
    {
        //todo
    }

    public void DisplayPoliceStation()
    {
        //todo
    }
}
```

UML Diagram

Administration class

```
- salaries : uint
- policeStationAddress : string
- equipmentPerStaion : byte
- equipmentType : string
- staffPerStation : byte

+ AdministrationOfPoliceStation() : void
+ CoordinatePoliceStations() : void
+ Transactions() : void
+ Documentation() : void
+ DisplayPoliceStation() : void
```

PoliceStation

Suspect class

Code

```
class Suspect
{
    private string suspecttID;
    private string suspectName;
    private string suspectAddress;
    private ulong suspectContactNumber;
    private byte suspectAge;

    public void AddSuspect()
    {
        //todo
    }

    public void DeleteSuspect()
    {
        //todo
    }

    public void CaseProceedings()
    {
        //todo
    }

    public void DisplaySuspect()
    {
        //todo
    }

    public void Documentation()
    {
        //todo
    }
}
```

UML Diagram

Suspect class

```
- suspecttID : string
- suspectName : string
- suspectAddress : string
- suspectContactNumber : ulong
- suspectAge : byte

+ AddSuspect() : void
+ DeleteSuspect() : void
+ CaseProceedings() : void
+ DisplaySuspect() : void
+ Documentation() : void
```