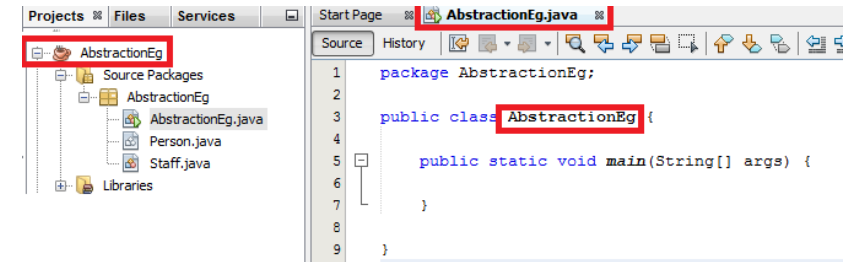


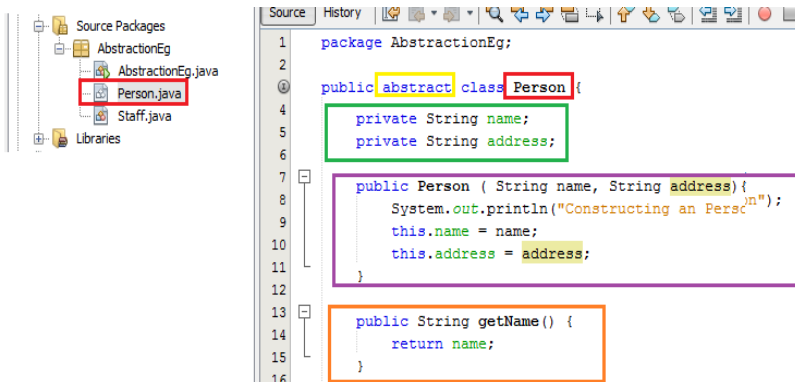
Abstraction Example

- Create NetBeans project **Abstraction**



The screenshot shows the NetBeans IDE with a project named 'AbstractionEg'. The project structure on the left includes 'Source Packages' containing 'AbstractionEg', which has sub-packages 'AbstractionEg.java', 'Person.java', and 'Staff.java'. The main editor window shows the code for 'AbstractionEg.java':

```
1 package AbstractionEg;
2
3 public class AbstractionEg {
4
5     public static void main(String[] args) {
6
7     }
8
9 }
```

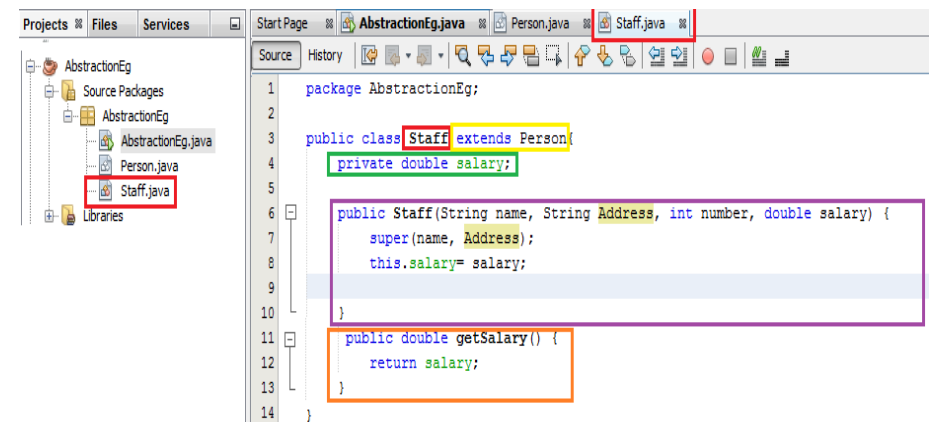


The screenshot shows the NetBeans IDE with the project structure on the left, highlighting 'Person.java'. The main editor window shows the code for 'Person.java':

```
1 package AbstractionEg;
2
3 public abstract class Person {
4     private String name;
5     private String address;
6
7     public Person ( String name, String address) {
8         System.out.println("Constructing an Persc\n");
9         this.name = name;
10        this.address = address;
11    }
12
13    public String getName() {
14        return name;
15    }
16 }
```

- Create a class **Person** within the **Abstract** package.
- Add class modifier **abstract** to the class declaration .
- Add two private variables **Name, Address**.
- Create the person method and include the code listed
- Create a **getname()** method and include the code listed.

- Create a class **Staff** within the **Abstract** package.
- Add class modifier **Extends person** to the class declaration.
- Add one private variables **salary**.
- Create the salary Method and include the code listed.
- Create a **getsalary()** Method and include the code listed



The screenshot shows the NetBeans IDE with the project structure on the left, highlighting 'Staff.java'. The main editor window shows the code for 'Staff.java':

```
1 package AbstractionEg;
2
3 public class Staff extends Person {
4     private double salary;
5
6     public Staff(String name, String Address, int number, double salary) {
7         super(name, Address);
8         this.salary= salary;
9     }
10
11     public double getSalary() {
12         return salary;
13     }
14 }
```

Abstraction Example

```
StartPage  AbstractionEg.java  Person.java  Staff.java
Source  History  [Icons]
1  package AbstractionEg;
2
3  public class AbstractionEg {
4
5      public static void main(String[] args) {
6          Staff s = new Staff("George W.", "Houston, TX", 43,4000);
7          s.setAddress("xxxx");
8          System.out.println(s.getName());
9          System.out.println(s.getSalary());
10         System.out.println(s.getAddress());
11     }
12
13 }
```

- Return to the AbstractEg class
- Instantiate the staff class and add the parameters listed
- Use println to out the values

Abstraction

```
package AbstractionEg;

public class AbstractionEg {

    public static void main(String[] args) {
        Staff s = new Staff("George W.", "Houston, TX", 43,4000);
        s.setAddress("xxxx");
        System.out.println(s.getName());
        System.out.println(s.getSalary());
        System.out.println(s.getAddress());
    }
}
```

```
package AbstractionEg;

public abstract class Person {
    private String name;
    private String address;

    public Person (String name, String address){
        System.out.println("Constructing an Person");
        this.name = name;
        this.address = address;
    }

    public String getName() {
        return name;
    }
}
```

```
package AbstractionEg;

public class Staff extends Person{
    private double salary;

    public Staff(String name, String Address, int number, double salary) {
        super(name, Address);
        this.salary= salary;
    }

    public double getSalary() {
        return salary;
    }

    public void setSalary(double salary) {
        this.salary = salary;
    }
}
```