

```
File Edit Selection View Go Run Terminal Help
TransportasiAir.java - MODUL2_FAQIH - Visual Studio Code

EXPLORER
src
  MainApp.java
  TransportasiAir.java
  Sampan.java
  Kapal.java
  .vscode
  bin
  lib
  src
    Kapal.java
    MainApp.java
    Sampan.java
    TransportasiAir.java
  README.md

OUTLINE
TIMELINE
JAVA PROJECTS
main*
24°C
Berawan

Ln 12, Col 57 Spaces: 4 UTF-8 CRLF {} Java Go Live Prettier R2 15:50 07/12/2022
```

```
public class TransportasiAir {
    protected int jumlahKursi;
    protected int biaya;

    public TransportasiAir(int jumlahKursi, int biaya) {
        this.jumlahKursi = jumlahKursi;
        this.biaya = biaya;
    }

    public void informasi() {
        System.out.println("Transportasi air dengan jenis yang tidak diketahui dengan kursi " + jumlahKursi
            + " ditetapkan dengan biaya sebesar Rp. " + biaya);
    }

    public void berlayar() {
        System.out.println("Transportasi air dengan jenis yang tidak diketahui sedang berlayar");
    }

    public void berlubu() {
        System.out.println("Transportasi air dengan jenis yang tidak diketahui berlubu di pantai");
    }
}
```

```
File Edit Selection View Go Run Terminal Help
Sampan.java - MODUL2_FAQIH - Visual Studio Code

EXPLORER
src
  MainApp.java
  TransportasiAir.java
  Sampan.java
  Kapal.java
  .vscode
  bin
  lib
  src
    Kapal.java
    MainApp.java
    Sampan.java
    TransportasiAir.java
  README.md

OUTLINE
TIMELINE
JAVA PROJECTS
main*
24°C
Berawan

Ln 12, Col 57 Spaces: 4 UTF-8 CRLF {} Java Go Live Prettier R2 15:50 07/12/2022
```

```
public class Sampan extends TransportasiAir {
    protected int layar;

    public Sampan(int jumlahKursi, int biaya, int layar) {
        super(jumlahKursi, biaya);
        this.layar = layar;
    }

    @Override
    public void informasi() {
        System.out.println("Transportasi air dengan jenis sampan dengan kursi berjumlah " + jumlahKursi
            + " ditetapkan dengan biaya sebesar Rp. " + biaya);
    }

    public void berlayar() {
        System.out.println("Transportasi air dengan jenis sampan sedang berlayar menggunakan " + layar + " layar");
    }

    public void berlubu() {
        System.out.println("Transportasi air dengan jenis sampan sedang berlubu dipantai tanpa jangkar");
    }

    public void berlubu(int jangkar) {
        System.out.println("Transportasi air dengan jenis sampan sedang berlubu menggunakan " + jangkar + " jangkar");
    }
}
```

This screenshot shows the Visual Studio Code editor with the `Kapal.java` file open. The class `Kapal` extends `TransportasiAir` and implements the `mesin` attribute. The methods defined are `informasi()`, `berlayar()`, `berlayar(int kecepatan)`, and `berlabu()`. The IDE interface includes the Explorer, Search, and Run and Debug panels. The status bar at the bottom shows the file is at line 12, column 57, with 4 spaces, UTF-8 encoding, and CRLF line endings.

```
public class Kapal extends TransportasiAir {
    protected String mesin;

    public Kapal(int jumlahkursi, int biaya, String mesin) {
        super(jumlahkursi, biaya);
        this.mesin = mesin;
    }

    @Override
    public void informasi() {
        System.out.println("Transportasi air dengan jenis kapal dengan kursi berjumlah " + jumlahkursi
            + " ditetapkan dengan biaya sebesar Rp. " + biaya);
    }

    public void berlayar() {
        System.out.println("Transportasi air dengan jenis kapal sedang berlayar menggunakan mesin " + mesin
            + " dengan kecepatan stabil di kisaran " + kecepatan + " knot");
    }

    public void berlayar(int kecepatan) {
        System.out.println("Transportasi air dengan jenis kapal sedang berlayar menggunakan mesin " + mesin
            + " dengan kecepatan stabil di kisaran " + kecepatan + " knot");
    }

    public void berlabu() {
        System.out.println("Transportasi air dengan jenis kapal berlabuh di pantai");
    }
}
```

This screenshot shows the Visual Studio Code editor with the `MainApp.java` file open. The class `MainApp` contains the `main` method, which creates instances of `TransportasiAir`, `Sampan`, and `Kapal`, and calls their `informasi()`, `berlayar()`, and `berlabu()` methods. The IDE interface includes the Explorer, Search, and Run and Debug panels. The status bar at the bottom shows the file is at line 20, column 26, with 4 spaces, UTF-8 encoding, and LF line endings.

```
public class MainApp {
    public static void main(String[] args) throws Exception {
        TransportasiAir transportasiAir1 = new TransportasiAir(jumlahkursi: 4, biaya: 20000);
        transportasiAir1.informasi();
        transportasiAir1.berlayar();
        transportasiAir1.berlabu();
        System.out.println(x: "");

        Sampan sampan1 = new Sampan(jumlahkursi: 20, biaya: 50000, layar: 2);
        sampan1.informasi();
        sampan1.berlayar();
        sampan1.berlabu();
        sampan1.berlabu(jangkai: 2);
        System.out.println(x: "");

        Kapal kapal1 = new Kapal(jumlahkursi: 50, biaya: 100000, mesin: "Diesel");
        kapal1.informasi();
        kapal1.berlayar();
        kapal1.berlayar(kecepatan: 20);
        kapal1.berlabu();
    }
}
```

The image shows a Windows desktop with a Java IDE (Visual Studio Code) open. The IDE has a dark theme. The main editor window displays a Java file named 'MainApp.java'. The code defines a 'MainApp' class with a 'main' method. Inside the 'main' method, there are several lines of code that create objects of 'Sampan' and 'Kapal' classes and call their methods. The 'Sampan' class has methods for 'berlayar()' and 'berlabuh()'. The 'Kapal' class has methods for 'berlayar()' and 'berlabuh()'. The output console on the right shows the execution results of the program. The status bar at the bottom indicates the file is 'Ln 20, Col 26' and the encoding is 'UTF-8'. The system tray at the bottom shows the date and time as '07/12/2022 15:51'.