a. Class(Sınıf) Nedir?

Sınıf, nesne yönelimli programlamada bir nesnenin özelliklerini (veri) ve davranışlarını (metotlar) tanımlayan bir şablondur.

b. Inheritance(Kalıtım) Nedir?

Kalıtım, bir sınıfın başka bir sınıfın özelliklerini ve metotlarını devralmasını sağlayan bir mekanizmadır. Kod tekrarını azaltır.

c. Encapsulation(Kapsülleme) Nedir?

Encapsulation, bir sınıfın özelliklerini(private) dışarıdan erişime kapatıp, yalnızca belirli metotlar(getter/setter) aracılığıyla erişim sağlama işlemidir. Veriyi korumak için kullanılır.

d. Polymorphism(Çok Biçimlilik) Nedir?

Polymorphism, aynı metot adının farklı alt sınıflarda farklı işlevselliklerle kullanılabilmesini sağlar. Örneğin, bir sesCikar metodu farklı hayvanlar için farklı sesler çıkarır.

e. Public ve Private Nedir?

public: Her yerden erişilebilir özellik/metot.

private: Yalnızca tanımlandığı sınıf içerisinden erişilebilir özellik/metot.

1-)  
public class Main {

public static void main(String[] args) {

int sayi = 5; --

System.out.println(sayi + "! = " + faktoriyelHesapla(sayi));}

public static int faktoriyelHesapla(int sayi) {return (sayi <= 1) ? 1 : sayi \* faktoriyelHesapla(sayi - 1);} }

2-)

public class Main {

public static void main(String[] args) {

int sayi = -3; --

System.out.println("Sayı: " + sayi + " -> " + sorgula(sayi));

}

public static String sorgula(int sayi) {

if (sayi > 0) {return "Pozitif";} else if (sayi < 0) {return "Negatif";} else {return "Sıfır";} } }

3-)

public class Main {

public static void main(String[] args) {

int sayi1 = 12, sayi2 = 5, sayi3 = 20; --

System.out.println("En küçük sayı: " + kucukHangisi(sayi1, sayi2, sayi3));

}public static int kucukHangisi(int a, int b, int c) {return Math.min(a, Math.min(b, c));} }

4-)

public class Main {

public static void main(String[] args) {

int sayi = 10; --

System.out.println("0'dan " + sayi + "'ye kadar olan çift sayılar:");

ciftSayilar(sayi);

} public static void ciftSayilar(int sayi) {

for (int i = 0; i <= sayi; i++) {if (i % 2 == 0) { System.out.println(i);} } } }

5-)

public class Main {

public static void main(String[] args) {

double x = 2; // Taban

int y = 3; // Kuvvet

System.out.println(x + " üzeri " + y + " = " + kuvvetAl(x, y));}

public static double kuvvetAl(double x, int y) {return Math.pow(x, y);} }

6-)

public class Main {

public static void main(String[] args) {

String ad = "Taha Can";

String soyad = "Demir"; // Örnek ad ve soyad

hosgeldinizMesaji(ad, soyad);}

public static void hosgeldinizMesaji(String ad, String soyad) {

System.out.println("Merhaba " + ad + " " + soyad + ", hoş geldiniz.");} }

7-)

public class Main {

public static void main(String[] args) {

int sayi = 29; --

if (asalMi(sayi)) {System.out.println(sayi + " asaldır.");}

else {System.out.println(sayi + " asal değildir.");} }

public static boolean asalMi(int sayi) {

if (sayi < 2) return false; // 2'den küçük sayılar asal değildir

for (int i = 2; i <= Math.sqrt(sayi); i++) {

if (sayi % i == 0) {return false; // Bölünebiliyorsa asal değildir} }

return true; // Asal} } }

8-)

Button btn= findViewById(R.id.myButton);

btn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {butonaTikla(); } }); }

public void butonaTikla() { Toast.makeText(this, "Butona tıklandı!", Toast.LENGTH\_SHORT).show();} }

9-)

public class Main {

public static void main(String[] args) {

int sayi1 = 10; // Birinci sayı

int sayi2 = 20; // İkinci sayı

int sonuc = topla(sayi1, sayi2); // Metodu çağır ve sonucu al

System.out.println("Toplam: " + sonuc);

}

public static int topla(int sayi1, int sayi2) {return sayi1 + sayi2;} }

11-)

public class Ogrenci { // Öğrenci Sınıfı

private String ad;

private String soyad;

private int yas;

private String sinif;

// Getter ve Setter metotları

public String getAd() {return ad; }

public void setAd(String ad) {this.ad = ad; }

public String getSoyad() {return soyad;}

public void setSoyad(String soyad) {this.soyad = soyad; }

public int getYas() {return yas; }

public void setYas(int yas) {this.yas = yas;}

public String getSinif() {return sinif;}

public void setSinif(String sinif) {this.sinif = sinif;} }

// MainActivity

public class MainActivity {

public static void main(String[] args) {

Ogrenci ogrenci = new Ogrenci();

ogrenci.setAd("Taha");

ogrenci.setSoyad("Demir");

ogrenci.setYas(16);

ogrenci.setSinif("11A");

// Bilgiler get metotlarıyla ekrana yazdırılıyor

System.out.println("Ad: " + ogrenci.getAd());

System.out.println("Soyad: " + ogrenci.getSoyad());

System.out.println("Yaş: " + ogrenci.getYas());

System.out.println("Sınıf: " + ogrenci.getSinif());} }

12-)

class Hayvan {

private String ad;

private String tur;

private int yas;

public String getAd() {return ad; }

public void setAd(String ad) {this.ad = ad;}

public String getTur() {return tur;}

public void setTur(String tur) {this.tur = tur;}

public int getYas() { return yas;}

public void setYas(int yas) {this.yas = yas;}

public void sesCikar() {System.out.println("Hayvan sesi");} }

class Kedi extends Hayvan {// Kedi Sınıfı

@Override

public void sesCikar() {System.out.println("Miyav");} }

class Kopek extends Hayvan { // Köpek Sınıfı

@Override

public void sesCikar() {System.out.println("Hav");} }

// MainActivity

public class MainActivity {

public static void main(String[] args) {

Kedi kedi = new Kedi(); // Kedi ve Köpek nesneleri oluşturuluyor

Kopek kopek = new Kopek();

kedi.setAd("Minnak"); // Nesneler için özellikler atanıyor

kedi.setTur("Kedi");

kedi.setYas(2);

kopek.setAd("Karabaş");

kopek.setTur("Köpek");

kopek.setYas(4);

System.out.println(kedi.getAd() + ": "); // sesCikar metotları çağrılarak sesler ekrana yazdırılıyor

kedi.sesCikar();

System.out.println(kopek.getAd() + ": ");

kopek.sesCikar();} }

13-)

// Araç Sınıfı

class Arac {

private String plaka;

private String model;

private int yukKapasitesi;

public String getPlaka() {

return plaka;

}

public void setPlaka(String plaka) {

this.plaka = plaka;

}

public String getModel() {

return model;

}

public void setModel(String model) {

this.model = model;

}

public int getYukKapasitesi() {

return yukKapasitesi;

}

public void setYukKapasitesi(int yukKapasitesi) {

this.yukKapasitesi = yukKapasitesi;

}

public void tasimaBilgisi() {

System.out.println("Bu araç yük taşıyor.");

}

}

// Kamyon Sınıfı

class Kamyon extends Arac {

@Override

public void tasimaBilgisi() {

System.out.println("Bu kamyon yük taşır.");

}

}

// Tır Sınıfı

class Tir extends Arac {

@Override

public void tasimaBilgisi() {

System.out.println("Bu tır büyük yük taşır.");

}

}

// MainActivity

public class MainActivity {

public static void main(String[] args) {

// Araç dizisi oluşturuluyor

Arac[] araclar = new Arac[2];

// Kamyon ve Tır nesneleri oluşturuluyor

Kamyon kamyon = new Kamyon();

kamyon.setPlaka("34ABC123");

kamyon.setModel("Mercedes");

kamyon.setYukKapasitesi(2000);

Tir tir = new Tir();

tir.setPlaka("06DEF456");

tir.setModel("Volvo");

tir.setYukKapasitesi(10000);

// Nesneler diziye ekleniyor

araclar[0] = kamyon;

araclar[1] = tir;

// Döngü ile araçların tasimaBilgisi metodu çağrılıyor

for (Arac arac : araclar) {

arac.tasimaBilgisi();

}

}

}

14-)

// Ürün Sınıfı

class Urun {

private String urunAdi;

private double fiyat;

public String getUrunAdi() {

return urunAdi;

}

public void setUrunAdi(String urunAdi) {

this.urunAdi = urunAdi;

}

public double getFiyat() {

return fiyat;

}

public void setFiyat(double fiyat) {

this.fiyat = fiyat;

}

}

// Enstrüman Sınıfı

class Enstruman extends Urun {

private String turu;

public String getTuru() {

return turu;

}

public void setTuru(String turu) {

this.turu = turu;

}

}

// MainActivity

public class MainActivity {

public static void main(String[] args) {

// Enstrüman nesnesi oluşturuluyor

Enstruman gitar = new Enstruman();

// Bilgiler set metotlarıyla giriliyor

gitar.setUrunAdi("Gitar");

gitar.setFiyat(1500.0);

gitar.setTuru("Akustik");

// Bilgiler get metotlarıyla ekrana yazdırılıyor

System.out.println("Ürün Adı: " + gitar.getUrunAdi());

System.out.println("Fiyat: " + gitar.getFiyat());

System.out.println("Türü: " + gitar.getTuru());

}

}

15-)

// Üye Sınıfı

class Uye {

private String ad;

private String soyad;

private int uyeNo;

public String getAd() {

return ad;

}

public void setAd(String ad) {

this.ad = ad;

}

public String getSoyad() {

return soyad;

}

public void setSoyad(String soyad) {

this.soyad = soyad;

}

public int getUyeNo() {

return uyeNo;

}

public void setUyeNo(int uyeNo) {

this.uyeNo = uyeNo;

}

}

// Kitap Sınıfı

class Kitap {

private String kitapAdi;

private String yazar;

private int yayinYili;

private boolean oduncDurumu; // true: ödünçte, false: mevcut

public String getKitapAdi() {

return kitapAdi;

}

public void setKitapAdi(String kitapAdi) {

this.kitapAdi = kitapAdi;

}

public String getYazar() {

return yazar;

}

public void setYazar(String yazar) {

this.yazar = yazar;

}

public int getYayinYili() {

return yayinYili;

}

public void setYayinYili(int yayinYili) {

this.yayinYili = yayinYili;

}

public boolean isOduncDurumu() {

return oduncDurumu;

}

public void setOduncDurumu(boolean oduncDurumu) {

this.oduncDurumu = oduncDurumu;

}

}

// Islem Sınıfı

class Islem {

private Uye uye;

private Kitap kitap;

private String islemTarihi;

public Uye getUye() {

return uye;

}

public void setUye(Uye uye) {

this.uye = uye;

}

public Kitap getKitap() {

return kitap;

}

public void setKitap(Kitap kitap) {

this.kitap = kitap;

}

public String getIslemTarihi() {

return islemTarihi;

}

public void setIslemTarihi(String islemTarihi) {

this.islemTarihi = islemTarihi;

}

public void islemDetayi() {

if (kitap.isOduncDurumu()) {

System.out.println("Kitap şu an ödünç alınamaz.");

} else {

System.out.println("Kitap ödünç alındı: [" + kitap.getKitapAdi() + "], Üye: [" + uye.getAd() + " " + uye.getSoyad() + "]");

}

}

}

// MainActivity

public class MainActivity {

public static void main(String[] args) {

// Üye oluşturuluyor

Uye uye = new Uye();

uye.setAd("Ahmet");

uye.setSoyad("Yılmaz");

uye.setUyeNo(101);

// Kitap oluşturuluyor

Kitap kitap = new Kitap();

kitap.setKitapAdi("Bilinmeyen Bir Kadının Mektubu");

kitap.setYazar("Stefan Zweig");

kitap.setYayinYili(1922);

kitap.setOduncDurumu(false); // Mevcut

// İşlem oluşturuluyor

Islem islem = new Islem();

islem.setUye(uye);

islem.setKitap(kitap);

islem.setIslemTarihi("07.01.2025");

// İşlem detayları yazdırılıyor

islem.islemDetayi();

}

}

16-)

import java.util.HashMap;

import java.util.Map;

public class MainActivity {

public static void main(String[] args) {

int[] arr = {1, 2, 3, 4, 5, 1, 2, 6, 7, 8, 9, 2};

Map<Integer, Integer> frequencyMap = new HashMap<>();

// Elemanların sıklığını belirleme

for (int num : arr) {

frequencyMap.put(num, frequencyMap.getOrDefault(num, 0) + 1);

}

// Tekrar eden elemanları yazdırma

System.out.println("Tekrar eden elemanlar:");

for (Map.Entry<Integer, Integer> entry : frequencyMap.entrySet()) {

if (entry.getValue() > 1) {

System.out.println(entry.getKey() + " - " + entry.getValue() + " kez tekrar ediyor.");

} } } }

17-)

public class MainActivity {

public static void main(String[] args) {

String[] arr = {"elma", "armut", "muz", "elma", "çilek", "muz"};

Map<String, Integer> frequencyMap = new HashMap<>();

// Elemanların sıklığını belirleme

for (String item : arr) {

frequencyMap.put(item, frequencyMap.getOrDefault(item, 0) + 1);

}

// Tekrar eden elemanları yazdırma

System.out.println("Tekrar eden elemanlar:");

for (Map.Entry<String, Integer> entry : frequencyMap.entrySet()) {

if (entry.getValue() > 1) {

System.out.println(entry.getKey() + " - " + entry.getValue() + " kez tekrar ediyor.");

}

}

}

}

18-)

import java.util.ArrayList;

import java.util.Collections;

public class MainActivity {

public static void main(String[] args) {

// ArrayList oluşturuluyor

ArrayList<Integer> list = new ArrayList<>();

list.add(45);

list.add(12);

list.add(89);

list.add(23);

list.add(56);

// En küçük ve en büyük elemanları bulma

int min = Collections.min(list);

int max = Collections.max(list);

// Sonuçları yazdırma

System.out.println("En küçük eleman: " + min);

System.out.println("En büyük eleman: " + max);

}

}

19-)

public class MainActivity {

public static void main(String[] args) {

int[] numbers = {10, 20, 30, 40, 50};

int toplam = 0;

// Dizideki elemanların toplamını hesaplama

for (int number : numbers) {

toplam += number;

}

// Ortalama hesaplama

double ortalama = (double) toplam / numbers.length;

// Sonuçları yazdırma

System.out.println("Dizideki eleman sayısı: " + numbers.length);

System.out.println("Dizideki elemanların toplamı: " + toplam);

System.out.println("Ortalama: " + ortalama);

}

}

20-)

public class MainActivity {

public static void main(String[] args) {

System.out.println("1'den 100'e kadar çift sayılar (For Döngüsü):");

for (int i = 2; i <= 100; i += 2) {System.out.print(i + " "); } } }

public class MainActivity {

public static void main(String[] args) {

System.out.println("1'den 100'e kadar çift sayılar (While Döngüsü):");

int i = 2;

while (i <= 100) {System.out.print(i + " "); i += 2;} } }

---------------------------------

public class MainActivity {

public static void main(String[] args) {

int faktoriyel = 1;

for (int i = 1; i <= 7; i++) { faktoriyel \*= i; }

System.out.println("7 sayısının faktöriyeli: " + faktoriyel);} }

public class MainActivity {

public static void main(String[] args) {

int faktoriyel = 1;

int i = 1;

while (i <= 7) { faktoriyel \*= i; i++; }

System.out.println("7 sayısının faktöriyeli: " + faktoriyel);} }

public class MainActivity {

public static void main(String[] args) {

int count = 0;

for (int i = 10; i <= 300; i++) {

if (i % 17 == 0 || i % 23 == 0) {count++;}

}

System.out.println("10 ile 300 arasında 17 veya 23'e tam bölünen sayıların adedi: " + count); } }

public class MainActivity {

public static void main(String[] args) {

int count = 0;

int i = 10;

while (i <= 300) {

if (i % 17 == 0 || i % 23 == 0) {count++; }

i++;}

System.out.println("10 ile 300 arasında 17 veya 23'e tam bölünen sayıların adedi: " + count);} }