**Vamos codificar:**

1) Crie uma classe cliente e apresente os atributos e métodos referentes esta classe, em seguida crie um objeto cliente, defina as instancias deste objeto e apresente as informações deste objeto no console.

package ContaBancaria;  
  
//nome do objeto  
public class ContaBancaria {  
  
 // atributos-caracteristicas do objeto  
 private int numero;  
 private int agencia;  
 private int tipo;  
 private String titular;  
 private float saldo;  
  
 //metodos construtores da classe Conta Bancaria  
  
 public ContaBancaria(int numero, int agencia, int tipo, String titular, float saldo) {  
 this.numero = numero;  
 this.agencia = agencia;  
 this.tipo = tipo;  
 this.titular = titular;  
 this.saldo = saldo;  
 }  
  
 public ContaBancaria() {}  
  
  
 // metodos acessores da classe Conta Bancaria - precisa ter em todos os objetos criados, pois sem eles no conseguimos nem visualizar as informaes, nem inserir novas informaes aos novos objetos criados a partir dessa classe.  
  
 public int getNumero() {  
 return numero;  
 }  
  
 public void setNumero(int numero) {  
 this.numero = numero;  
 }  
  
 public int getAgencia() {  
 return agencia;  
 }  
  
 public void setAgencia(int agencia) {  
 this.agencia = agencia;  
 }  
  
 public int getTipo() {  
 return tipo;  
 }  
  
 public void setTipo(int tipo) {  
 this.tipo = tipo;  
 }  
  
 public String getTitular() {  
 return titular;  
 }  
  
 public void setTitular(String titular) {  
 this.titular = titular;  
 }  
  
 public float getSaldo() {  
 return saldo;  
 }  
  
 public void setSaldo(float saldo) {  
 this.saldo = saldo;  
 }  
  
 //metodos especficos de conta bancaria - depende de objeto para objeto, e temos que levar em considerao sempre qual o objeto a ser criado, e criar mtodos especficos de acordo com a funcionalidade de cada objeto  
 //aqui criamos o metodo sacar, que recebe como parametro um valor a ser sacado. ele precisa ter obrigatriamente o mesmo tipo de dado do saldo, que o valor que vai ser modificado a partir do saque, no caso os dois precisam ser float  
 public boolean sacar(float valorSacado) {  
 // antes de tudo eu fao uma verificao,pegando o saldo da minha conta a partir do getSaldo, e comparando se o valor do saldo maior que o valor a ser sacado. se no for eu j barro o usuario na hora, indicando que o saldo insuficiente e retornando um false na funo, mostrando que o saque no foi realizado  
 if(this.getSaldo() < valorSacado) {  
 System.*out*.println("Saldo insuficiente, tente quando estiver rico");  
 return false;  
 }  
  
 // caso o valor sacado for menor que o do saldo, eu fao ento a operao de saque, pegando novamente o valor do saldo, e subtraindo o valor do saque  
  
 this.setSaldo(this.getSaldo() - valorSacado);  
 // por ultimo eu mostro ao meu usurio o valor sacado, e quando sobrou na conta, retornando true, o saque foi feito  
 System.*out*.println("Valor de" + valorSacado + " sacado com sucesso! O seu valor de saldo agora "+ this.getSaldo());  
 return true;  
 }  
}

=== agora vamos testar===

///Este é o construtor da ContaBancaria  
  
package ContaBancaria;  
  
public class TesteContaBancaria {  
  
 public static void main(String[] args) {  
  
 // aqui estamos instanciando um novo objeto de conta bancaria usando a classe conta bancaria. Usando o m*todo construtor, passamos todos os dados para preecher os atributos dessa nova conta gerada* ContaBancaria novaConta = new ContaBancaria(12, 65, 2, "Carol", 1200f);  
  
  
 // cada system.out.print est retornando os dados da nova conta criada, usando os metodos acessores get, lembrando que get para mostrar os dados, e set para inserir os dado.  
 System.*out*.println("Numero da Conta "+ novaConta.getNumero());  
 System.*out*.println("Numero da Agencia "+ novaConta.getAgencia());  
 System.*out*.println("Numero do Tipo "+ novaConta.getTipo());  
 System.*out*.println("Numero da Titular "+ novaConta.getTitular());  
 System.*out*.println("Numero da Saldo "+ novaConta.getSaldo());  
  
  
 // aqui estamos testando o metodo especifico criado para a classe de conta, que o sacar, informando diretamente o valor a ser sacado da conta criada.  
 novaConta.sacar(650);  
  
 }  
  
}

2) Crie uma classe avião e apresente os atributos e métodos referentes esta classe, em seguida crie um objeto avião, defina as instancias deste objeto e apresente as informações deste objeto no console.

package exerciciosPOO;  
/// nome da classe/objeto sempre maiúsculo  
public class Aviao {  
  
 /// atributos-caracteristicas do objeto  
 private String companhia;  
 private double peso;  
 private int velocidade;  
 private String cor;  
 private int lotacao;  
 private String nome;  
 private double preco;  
  
 ///contrutores gerados automaticamente  
 public Aviao (String companhia, double peso, int velocidade, String cor, int lotacao, String nome, double preco) {  
 this.companhia = companhia;  
 this.peso = peso;  
 this.velocidade = velocidade;  
 this.cor = cor;  
 this.lotacao = lotacao;  
 this.nome = nome;  
 this.preco = preco;  
 }  
 /// esse construtor deve ficar assim vazio  
 public Aviao () { }  
  
 /// getters and setters gerados automaticamente  
 public String getCompanhia() {  
 return companhia;  
 }  
  
 public void setCompanhia(String companhia) {  
 this.companhia = companhia;  
 }  
  
 public double getPeso() {  
 return peso;  
 }  
  
 public void setPeso(double peso) {  
 this.peso = peso;  
 }  
  
 public int getVelocidade() {  
 return velocidade;  
 }  
  
 public void setVelocidade(int velocidade) {  
 this.velocidade = velocidade;  
 }  
  
 public String getCor() {  
 return cor;  
 }  
  
 public void setCor(String cor) {  
 this.cor = cor;  
 }  
  
 public int getLotacao() {  
 return lotacao;  
 }  
  
 public void setLotacao(int lotacao) {  
 this.lotacao = lotacao;  
 }  
  
 public String getNome() {  
 return nome;  
 }  
  
 public void setNome(String nome) {  
 this.nome = nome;  
 }  
  
 public double getPreco() {  
 return preco;  
 }  
  
 public void setPreco(double preco) {  
 this.preco = preco;  
 }  
}

===agora vamos testar===

package exerciciosPOO;  
public class TesteAviao {  
 public static void main(String[] args) {  
 Aviao comercial = new Aviao("Gol", 2, 150, "branca", 500, "F420", 500);  
  
 System.*out*.println("A compania desse avião é: " + comercial.getCompanhia());  
 System.*out*.println("O peso desse avião é de: " + comercial.getPeso() + " toneladas");  
 System.*out*.println("A velocidade desse avião é de: " + comercial.getVelocidade() + " km/h");  
 System.*out*.println("A cor desse avião é: " + comercial.getCor());  
 System.*out*.println("A lotação desse avião é de: " + comercial.getLotacao() + " passageiros");  
 System.*out*.println("A nome desse avião é: " + comercial.getNome());  
 System.*out*.println("O preço desse avião é de: " + comercial.getPreco() + " R$");  
 }  
}

3) Crie uma classe produto eletrônico e apresente os atributos e métodos referentes esta classe, em seguida crie um objeto produto eletrônico, defina as instancias deste objeto e apresente as informações deste objeto no console.

4) Crie uma classe funcionário e apresente os atributos e métodos referentes esta classe, em seguida crie um objeto funcionário, defina as instancias deste objeto e apresente as informações deste objeto no console.

5) Crie uma classe patinete e apresente os atributos e métodos referentes esta classe, em seguida crie um objeto patinete, defina as instancias deste objeto e apresente as informações deste objeto no console.

6) Crie uma classe conta bancaria e apresente os atributos e métodos referentes esta classe, em seguida crie um objeto conta bancaria, defina as instancias deste objeto e apresente as informações deste objeto no console.

7) Crie uma classe paciente e apresente os atributos e métodos referentes esta classe, em seguida crie um objeto paciente, defina as instancias deste objeto e apresente as informações deste objeto no console.