

Classes e Objetos

Questão 1: Classes e Objetos: Crie uma classe abstrata chamada "Pessoa" que represente um ser humano. Pessoa deve ter atributos como "cpf", "nome" e "idade". Implemente os métodos para definir e obter esses atributos. Crie uma classe "Disciplina" com os atributos como "código", "nome" e "semestre".

```
minha.universidade.grupo.02 - Pessoa.java

1  public abstract class Pessoa {
2
3      private String cpf;
4      public String nome;
5      private int idade;
6
7      // Construtor
8      public Pessoa(String cpf, String nome, int idade) {
9          this.cpf = cpf;
10         this.nome = nome;
11         this.idade = idade;
12     }
13
14     // Método personalizado - aniversário
15     public void fazerAniversario(){
16         this.setIdade(this.getIdade() + 1);
17     }
18
19     // Getters
20     public String getCpf() {
21         return cpf;
22     }
23
24     public String getNome() {
25         return nome;
26     }
27
28     public int getIdade() {
29         return idade;
30     }
31
32     // Setters
33     public void setCpf(String cpf) {
34         this.cpf = cpf;
35     }
36
37     public void setNome(String nome) {
38         this.nome = nome;
39     }
40
41     public void setIdade(int idade) {
42         this.idade = idade;
43     }
44 }
45
```



minha.universidade.grupo.02 - Disciplina.java

```
1  public class Disciplina {
2
3      private String codigo;
4      private String nome;
5      private int semestre;
6
7      // Construtor
8      public Disciplina(String codigo, String nome, int semestre) {
9          this.codigo = codigo;
10         this.nome = nome;
11         this.semestre = semestre;
12     }
13
14     // Getters
15     public String getCodigo() {
16         return codigo;
17     }
18
19     public String getNome() {
20         return nome;
21     }
22
23     public int getSemestre() {
24         return semestre;
25     }
26
27     // Setters
28     public void setCodigo(String codigo) {
29         this.codigo = codigo;
30     }
31
32     public void setNome(String nome) {
33         this.nome = nome;
34     }
35
36     public void setSemestre(int semestre) {
37         this.semestre = semestre;
38     }
39
40     // Sobrescrita do método toString
41     @Override
42     public String toString(){
43         return nome;
44     }
45 }
46
```

Encapsulamento e Métodos

Questão 2: Encapsulamento e Métodos: Evolua a classe "Pessoa" adicionando um método público para incrementar a idade da pessoa. Crie um método público chamado "fazerAniversario".

```
minha.universidade.grupo.02 - Pessoa.java

1  public abstract class Pessoa {
2
3      private String cpf;
4      public String nome;
5      private int idade;
6
7      // Construtor
8      public Pessoa(String cpf, String nome, int idade) {
9          this.cpf = cpf;
10         this.nome = nome;
11         this.idade = idade;
12     }
13
14     // Método personalizado - aniversário
15     public void fazerAniversario(){
16         this.setIdade(this.getIdade() + 1);
17     }
18
19     // Getters
20     public String getCpf() {
21         return cpf;
22     }
23
24     public String getNome() {
25         return nome;
26     }
27
28     public int getIdade() {
29         return idade;
30     }
31
32     // Setters
33     public void setCpf(String cpf) {
34         this.cpf = cpf;
35     }
36
37     public void setNome(String nome) {
38         this.nome = nome;
39     }
40
41     public void setIdade(int idade) {
42         this.idade = idade;
43     }
44 }
45
```

Herança e Polimorfismo

Questão 3: Herança e Polimorfismo: Crie 3 classes derivadas de "Pessoa", são elas: "Visitante", "Aluno" e "Professor". Adicione um atributo "matrícula" para a classe "Aluno" e um método "pagarMensalidade". Adicione um atributo "centro" para a classe "Professor" e um método "darAula". Crie 2 classes derivadas de "Aluno", são elas: "Bolsista" e "Regular" e faça uma sobreposição no método "pagarMensalidade" para ambas as classes.

```
minha.universidade.grupo.02 - Visitante.java

1  public class Visitante extends Pessoa {
2
3      // Construtor
4      public Visitante(String cpf, String nome, int idade) {
5          super(cpf, nome, idade);
6      }
7  }
```

```
minha.universidade.grupo.02 - Aluno.java

1  public class Aluno extends Pessoa {
2
3      private String matricula;
4
5      // Construtor
6      public Aluno(String cpf, String nome, int idade, String matricula) {
7          super(cpf, nome, idade);
8          this.matricula = matricula;
9      }
10
11     // Getters e Setters para matricula
12
13     public String getMatricula() {
14         return matricula;
15     }
16
17     public void setMatricula(String matricula) {
18         this.matricula = matricula;
19     }
20
21     // Método pagarMensalidade (VERIFICAR)
22     public void pagarMensalidade() {
23         System.out.println("Este aluno: " + nome + ", pagou a mensalidade.");
24     }
25
26     /* Sobrescrita do método toString. Ele é chamado automaticamente.
27     Serve para obter uma representação String de um objeto. Aqui ele
28     é usado para listar o nome dos alunos na lista de alunos criada na classe Turma*/
29     @Override
30     public String toString(){
31         return nome;
32     }
33 }
34
```

```
minha.universidade.grupo.02 - Professor.java

1  public class Professor extends Pessoa {
2
3      public String centro;
4
5      // Construtor
6      public Professor(String cpf, String nome, int idade, String centro) {
7          super(cpf, nome, idade);
8          this.centro = centro;
9      }
10
11     // Getters e Setters para centro
12
13     public String getCentro() {
14         return centro;
15     }
16
17     public void setCentro(String centro) {
18         this.centro = centro;
19     }
20
21     // Método para dar aula.
22     public void darAula(){
23         System.out.println("O professor " + nome + " dá aulas em " + centro);
24     }
25     @Override
26     public String toString(){
27         return nome;
28     }
29 }
30
```

```
minha.universidade.grupo.02 - Bolsista.java

1  public class Bolsista extends Aluno {
2
3      // Construtor
4      public Bolsista(String cpf, String nome, int idade, String matricula) {
5          super(cpf, nome, idade, matricula);
6      }
7
8      // Sobreposição do método pagarMensalidade
9      @Override
10     public void pagarMensalidade() {
11         System.out.println("O Aluno " + nome + " é Bolsista. Tem bolsa e não paga mensalidade.");
12     }
13 }
14
```

```
minha.universidade.grupo.02 - Regular.java

1  public class Regular extends Aluno {
2
3      // Construtor
4      public Regular(String cpf, String nome, int idade, String matricula) {
5          super(cpf, nome, idade, matricula);
6      }
7
8      // Sobreposição do método pagarMensalidade
9      @Override
10     public void pagarMensalidade() {
11         super.pagarMensalidade(); // Chamada do método da classe base
12     }
13 }
14
```

Associação e Composição

Questão 4: Associação e Composição: Crie uma classe, chamada "Turma". Adicione em turma os atributos "código", "disciplina", "professor" e "alunos", crie os métodos "adicionarAluno", "removerAluno" e "listarAlunos" e os implemente utilizando ArrayList. Cada atributo será do tipo da classe de mesmo nome, exceto alunos que terá uma lista de objetos do tipo "Aluno".

```
minha.universidade.grupo.02 - Turma.java

1  import java.util.ArrayList; // Biblioteca Java Util ArrayList
2
3  public class Turma {
4
5      private String codigo;
6      private Disciplina disciplina;
7      private Professor professor;
8      private ArrayList<Aluno> alunos;
9
10     // Construtor
11     public Turma(String codigo, Disciplina disciplina, Professor professor) {
12         this.codigo = codigo;
13         this.disciplina = disciplina;
14         this.professor = professor;
15         this.alunos = new ArrayList<>();
16     }
17     // Métodos para adc e remover aluno
18     public void adicionarAluno(Aluno aluno){
19         alunos.add(aluno);
20     }
21     public void removerAluno(Aluno aluno){
22         alunos.remove(aluno);
23         System.out.println("Aluno(a) " + aluno + " removido da disciplina " + disciplina + ".");
24     }
25     // Método para exibir as informações da turma
26     public void status(){
27         System.out.println("Código: " + codigo);
28         System.out.println("Professor: " + professor);
29         System.out.println("Disciplina: " + disciplina);
30         System.out.println("Alunos: " + alunos);
31     }
32
33     // Getters e Setters
34     public String getCodigo() {
35         return codigo;
36     }
37
38     public void setCodigo(String codigo) {
39         this.codigo = codigo;
40     }
41
42     public Disciplina getDisciplina() {
43         return disciplina;
44     }
45
46     public void setDisciplina(Disciplina disciplina) {
47         this.disciplina = disciplina;
48     }
49
50     public Professor getProfessor() {
51         return professor;
52     }
53
54     public void setProfessor(Professor professor) {
55         this.professor = professor;
56     }
57 }
58
```