

Interfaz ejemplo, valía cualquiera:

```
public interface Estudian {  
    public boolean tieneTodoAprobado();  
}
```

MiPersona:

```
private String id;  
private String nombre;  
public MiPersona() {  
    // TODO Apéndice de constructor generado automáticamente  
}  
public MiPersona(String id, String nombre) {  
    this.id = id;  
    this.nombre = nombre;  
}  
public String getId() {  
    return id;  
}  
public void setId(String id) {  
    this.id = id;  
}  
public String getNombre() {  
    return nombre;  
}  
public void setNombre(String nombre) {  
    this.nombre = nombre;  
}  
@Override  
public String toString() {  
    return "MiPersona [id=" + id + ", nombre=" + nombre + "];"  
}  
}
```

Estudiante:

```

public class Estudiante extends MiPersona implements Estudian{

    private String curso;
    private HashMap<String, Double> notas = new HashMap<>();

    public Estudiante(String id, String nombre, String curso) {
        super(id, nombre);
        this.curso = curso;
    }

    public boolean tieneTodoAprobado() {

        for (String asignatura: this.notas.keySet()) {
            if (this.notas.get(asignatura)<5) {
                return false;
            }
        }
        return true;
    }

    public boolean addNota(String asignatura, Double nota) {
        if (asignatura!=null && nota!=null) {
            notas.put(asignatura, nota);
            return true;
        }
        return false;
    }

    public String getCurso() {
        return curso;
    }

    public void setCurso(String curso) {
        this.curso = curso;
    }

    public HashMap<String, Double> getNotas() {
        return notas;
    }

    public void setNotas(HashMap<String, Double> notas) {
        this.notas = notas;
    }

    @Override
    public String toString() {
        return super.toString() + "curso=" + curso + ", notas=" + notas + "];
    }
}

```

GestionEstudiante

```

public class GestionEstudiantes {

    private List<Estudiante> estudiantes = new ArrayList<>();

    public GestionEstudiantes() {
        String [] asignaturas = {"FOL", "BBDD", "EntornoD", "Programación", "LMSGI"};
        for (int i = 1; i <=10; i++) {
            Estudiante e = new Estudiante("0000"+i, "Estu"+i, "DAW1");
            for (String asignatura : asignaturas) {
                Random rnd = new Random();
                e.addNota(asignatura, rnd.nextInt(10)+1.0);
            }
            estudiantes.add(e);
        }
    }

    public List<Estudiante> getEstudiantes(){
        return estudiantes;
    }

    public void addEstudiante(Estudiante e) {
        if (e!=null)
            estudiantes.add(e);
    }

    public Estudiante getEstudiante(String id) {
        for (Estudiante estudiante : estudiantes) {
            if (estudiante.getId().equalsIgnoreCase(id))
                return estudiante;
        }
        return null;
    }

    public HashMap<String, Double> getNotasEstudiante(String id){
        Estudiante estudiante = this.getEstudiante(id);
        return estudiante==null? null: estudiante.getNotas();
    }

    public HashMap<String, String> getAprobados(){
        HashMap<String, String> aprobados = new HashMap<>();
        for (Estudiante estudiante : estudiantes) {
            if (estudiante.tieneTodoAprobado()) {
                aprobados.put(estudiante.getId(), estudiante.getNombre());
            }
        }
        return aprobados;
    }

    public List<String> getAprobadosPorAsignatura(String asignatura){
        List<String> listadoAprobados = new ArrayList<>();
        for (Estudiante estudiante : estudiantes) {
            if (estudiante.getNotas().get(asignatura)>=5)
                listadoAprobados.add(estudiante.getNombre());
        }
        return listadoAprobados;
    }
}

```

POSIBLE APP:

```

public class App {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        GestionEstudiantes gestion = new GestionEstudiantes();

        for (Estudiante e : gestion.getEstudiantes()) {
            System.out.println(e.toString());
        }

        System.out.println("Estudiantes con todo aprobado");
        System.out.println("*****");
        HashMap<String, String> todoApro= gestion.getAprobados();
        for (String key : todoApro.keySet()) {
            System.out.println(key + "--" + todoApro.get(key) );
        }

        System.out.println("*****");
        System.out.println("Aprobados en FOL");
        for (String nombreEstu : gestion.getAprobadosPorAsignatura("FOL")) {
            System.out.println(nombreEstu);
        }
    }
}

```