

Team 16  
Nathan Lai  
Githel Lynn Suico

## EmployeeInfo.java

```
/**
 * Team 16
 * Githel Lynn Suico
 * Nathan Lai
 * Interface to store constant values for employee parent
 */
public interface EmployeeInfo {
    double FACULTY_MONTHLY_SALARY = 5000.00;
    int STAFF_MONTHLY_HOURS_WORKED = 160;
}
```

## Employee.java

```
/**
 * Team 16
 * Githel Lynn Suico
 * Nathan Lai
 * Employee class meant to be the parent for all staff and faculty types in the school
 */
public abstract class Employee implements EmployeeInfo{
    private String lastName;
    private String firstName;
    private String ID;

    /**
     * Default constructor
     */
    public Employee(){
        lastName = "n/a";
        firstName = "n/a";
        ID = "n/a";
    }

    /**
     * Argument constructor
     * @param lastName is the last name
     * @param firstName is the first name
     * @param ID is the employee number
     */
}
```

Team 16

Nathan Lai

Githel Lynn Suico

```
*/
public Employee(String lastName, String firstName, String ID){
    this.lastName = lastName;
    this.firstName = firstName;
    this.ID = ID;
}

/**
 * @return Summary of employee's details
 */
@Override
public String toString(){
    return "Last name: " + lastName + "\nFirst Name: " + firstName + "\nID: " + ID;
}

/**
 * @return Last name of employee
 */
public String getLastName(){
    return lastName;
}

/**
 * @return First name of employee
 */
public String getFirstName(){
    return firstName;
}

/**
 * @return Employee ID of employee
 */
public String getID(){
    return ID;
}

/**
 * @param lastName Last name to set to employee
 */
public void setLastName(String lastName) {
    this.lastName = lastName;
}
```

Team 16  
Nathan Lai  
Githel Lynn Suico

```
/**
 * @param firstName First name to set to employee
 */
public void setFirstName(String firstName){
    this.firstName = firstName;
}

/**
 * @param ID ID to set to employee
 */
public void setID(String ID) {
    this.ID = ID;
}

/**
 * Calculates the amount of money earned in a month
 * @return the amount earned in a month
 */
abstract public double monthlyEarning();
}
```

## Faculty.java

```
/**
 * Team 16
 * Githel Lynn Suico
 * Nathan Lai
 * Faculty class is child of employee, represents a faculty member
 */
public class Faculty extends Employee {
    public enum Level {
        AS, AO, FU
    }
    private Level facultyLevel;

    private Education education;

    /**
     * Default constructor
     */
}
```

Team 16

Nathan Lai

Githel Lynn Suico

```
public Faculty(){
    super();
    this.facultyLevel = null;
    education = new Education();
}

/**
 * Arguments except education constructor
 * @param lastName Last name of faculty
 * @param firstName First name of faculty
 * @param ID ID of faculty
 * @param facultyLevel Level of faculty
 */
public Faculty(String lastName, String firstName, String ID, Level facultyLevel){
    super(lastName, firstName, ID);
    this.facultyLevel = facultyLevel;
    education = new Education();
}

/**
 * Full arguments constructor
 * @param lastName Last name of faculty
 * @param firstName First name of faculty
 * @param ID ID of faculty
 * @param facultyLevel Level of faculty
 * @param major Major of faculty
 * @param research Number of researches done by faculty
 */
public Faculty(String lastName, String firstName, String ID, String degree, Level facultyLevel, String major,
int research){
    super(lastName, firstName, ID);
    this.facultyLevel = facultyLevel;
    education = new Education(degree, major, research);
}

/**
 * Calculates amount of money earned per month depending on faculty level
 * @return Amount of money earned in a month
 */
public double monthlyEarning(){
    double earnings = 0;
    switch(facultyLevel){
```

Team 16

Nathan Lai

Githel Lynn Suico

```
        case AS:
            earnings = FACULTY_MONTHLY_SALARY;
            break;
        case AO:
            earnings = 1.5 * FACULTY_MONTHLY_SALARY;
            break;
        case FU:
            earnings = 2 * FACULTY_MONTHLY_SALARY;
            break;
    }
    return earnings;
}

/**
 * @return Summary of faculty's details
 */
@Override
public String toString(){
    String description = super.toString();
    switch(facultyLevel){
        case AS:
            description += "\nLevel: Assistant";
            break;
        case AO:
            description += "\nLevel: Associate";
            break;
        case FU:
            description += "\nLevel: Full";
            break;
    }
    description += "\nDegree: " + education.getDegree() +
        "\nMajor: " + education.getMajor() +
        "\nResearch: " + education.getResearch();
    return description;
}
}
```

## Education.java

```
/**
 * Team 16
 * Githel Lynn Suico
```

Team 16

Nathan Lai

Githel Lynn Suico

\* Nathan Lai

\* Stores information about an employee's education

\*/

```
public class Education {
    private String degree;
    private String major;
    private int research;

    /**
     * Default constructor
     */
    public Education(){
        degree = "No Degree";
        major = "Undeclared";
        research = 0;
    }

    /**
     * Arguments constructor
     * @param degree is MS or PhD
     * @param major is the major
     * @param research is the number of researches
     */
    public Education(String degree, String major, int research){
        this.degree = degree;
        this.major = major;
        this.research = research;
    }

    /**
     * @return Degree owned
     */
    public String getDegree() {
        return degree;
    }

    /**
     * @return Current major
     */
    public String getMajor() {
        return major;
    }
}
```

Team 16  
Nathan Lai  
Githel Lynn Suico

```
/**
 * @return Number of research projects participated in
 */
public int getResearch() {
    return research;
}

/**
 * @param degree Degree to set to
 */
public void setDegree(String degree) {
    this.degree = degree;
}

/**
 * @param major College major to set to
 */
public void setMajor(String major) {
    this.major = major;
}

/**
 * @param research Number of research projects to set to
 */
public void setResearch(int research) {
    this.research = research;
}
}
```

## Staff.java

```
/**
 * Team 16
 * Githel Lynn Suico
 * Nathan Lai
 * Represents a full time staff member
 */
public class Staff extends Employee {
    private double hourlyRate;

    /**
```

Team 16

Nathan Lai

Githel Lynn Suico

```
* Default constructor
```

```
*/
```

```
public Staff(){  
    super();  
    hourlyRate = 0;  
}
```

```
/**
```

```
* Arguments constructor
```

```
* @param lastName is last name of the staff member
```

```
* @param firstName is first name of the staff member
```

```
* @param ID is the employee identification number
```

```
*/
```

```
public Staff(String lastName, String firstName, String ID, double hourlyRate){  
    super(lastName, firstName, ID);  
    this.hourlyRate = hourlyRate;  
}
```

```
/**
```

```
* @return The hourly rate the staff is paid
```

```
*/
```

```
public double getHourlyRate() {  
    return hourlyRate;  
}
```

```
/**
```

```
* @param hourlyRate The hourly rate to set to the staff
```

```
*/
```

```
public void setHourlyRate(double hourlyRate) {  
    this.hourlyRate = hourlyRate;  
}
```

```
/**
```

```
* @return Monthly earnings, hourly rate times full time hours (160)
```

```
*/
```

```
public double monthlyEarning(){  
    return hourlyRate * STAFF_MONTHLY_HOURS_WORKED;  
}
```

```
/**
```



Team 16

Nathan Lai

Githel Lynn Suico

```
* @return Summary of staff's details
*/
@Override
public String toString(){
    return super.toString() + "\nHourly rate: $" + hourlyRate;
}
}
```

## PartTime.java

```
/**
 * Team 16
 * Githel Lynn Suico
 * Nathan Lai
 * Represents a part time staff member
 */
public class PartTime extends Staff {
    private int hoursWorked;

    /**
     * Default constructor
     */
    public PartTime(){
        super();
        this.hoursWorked = 0;
    }

    /**
     * Arguments constructor
     * @param lastName Last name of employee
     * @param firstName First name of employee
     * @param ID ID of employee
     * @param hoursWorked Number of hours worked by employee
     */
    public PartTime(String lastName, String firstName, String ID, double hourlyRate,int hoursWorked){
        super(lastName, firstName, ID, hourlyRate);
        this.hoursWorked = hoursWorked;
    }

    /**
     * @return Number of hours worked by employee
     */
}
```

Team 16

Nathan Lai

Githel Lynn Suico

```
    public int getHoursWorked() {
        return hoursWorked;
    }

    /**
     * @param hoursWorked Number of hours that employee will work
     */
    public void setHoursWorked(int hoursWorked) {
        this.hoursWorked = hoursWorked;
    }

    /**
     * Returns the monthly earnings, which is hourly rate multiplied by hours worked in 4 weeks
     * @return monthly earnings
     */
    public double monthlyEarning(){
        return getHourlyRate() * hoursWorked;
    }

    /**
     * @return Summary of part timers's details
     */
    @Override
    public String toString(){
        String description = super.toString();
        description += "\nHourly rate: " + getHourlyRate() +
            "\nHours worked per week: " + getHoursWorked();
        return description;
    }
}
```

## Main.java

```
public class Main {
    public static void main(String[] args){
        //Testing Staff class
        Staff allen = new Staff("Allen", "Paita", "123", 50.00);
        Staff zapata = new Staff("Zapata", "Steven", "456", 35.00);
        Staff rios = new Staff("Rios", "Enrique", "789", 40.00);

        System.out.println(allen);
    }
}
```

Team 16

Nathan Lai

Githel Lynn Suico

```
        System.out.println(zapata);
        System.out.println(rios);
        System.out.println();
        //Testing Faculty class
        Faculty johnson = new Faculty("Johnson", "Anne", "243", "Ph.D", Faculty.Level.FU, "Engineering", 3);
        Faculty bouris = new Faculty("Bouris", "William", "791", "Ph.D", Faculty.Level.AS, "English", 1);
        Faculty andrade = new Faculty("Andrade", "Christopher", "623", "MS", Faculty.Level.AS, "Physical
Education", 0);

        System.out.println(johnson);
        System.out.println(bouris);
        System.out.println(andrade);
        System.out.println();

        //Testing PartTime class
        PartTime guzman = new PartTime("Guzman", "Augusto", "455", 35.0, 30);
        PartTime depirro = new PartTime("Depirro", "Martin", "678", 30.0, 15);
        PartTime aldaco = new PartTime("Aldaco", "Marque", "945", 20.0, 35);

        System.out.println(guzman);
        System.out.println(bouris);
        System.out.println(andrade);
        System.out.println();
    }
}
```

Team 16  
Nathan Lai  
Githel Lynn Suico

## Runtime Output

```
D:\AdoptOpenJDK\bin\java.exe "-javaagent:D:\IntelliJ\Ultimate\IntelliJ
Last name: Allen
First Name: Paita
ID: 123
Hourly rate: $50.0
Last name: Zapata
First Name: Steven
ID: 456
Hourly rate: $35.0
Last name: Rios
First Name: Enrique
ID: 789
Hourly rate: $40.0

Last name: Johnson
First Name: Anne
ID: 243
Level: Full
Degree: Ph.D
Major: Engineering
Research: 3
Last name: Bouris
First Name: William
ID: 791
Level: Assistant
Degree: Ph.D
Major: English
Research: 1
```

Team 16  
Nathan Lai  
Githel Lynn Suico

```
Last name: Andrade  
First Name: Christopher  
ID: 623  
Level: Assistant  
Degree: MS  
Major: Physical Education  
Research: 0
```

```
Last name: Guzman  
First Name: Augusto  
ID: 455  
Hourly rate: $35.0  
Hourly rate: 35.0  
Hours worked per week: 30
```

```
Last name: Bouris  
First Name: William  
ID: 791  
Level: Assistant  
Degree: Ph.D  
Major: English  
Research: 1
```

```
Last name: Andrade  
First Name: Christopher  
ID: 623  
Level: Assistant  
Degree: MS  
Major: Physical Education  
Research: 0
```

```
Process finished with exit code 0
```