Matthew Brandao, Armit Patel
CS213 - Project #3 Test Case Document

Test Case #	**Purpose of the test cases**	**Input Data**	**Expected Output**
1	Test the parameterized constructor for **Parttime** class with all completely valid credentials. The constructor requires the name, department, date hired, and hourly rate input data. Radio buttons from the GUI are used to collect certain data from the user.	**Instance #1:** “Smith, Joe”, CS radio button selected, “3/09/2021”, “37.50”, Part time radio button selected	The GUI successfully displays a message saying the employee was added to the database. Add employee method returns true.
2	Test the parameterized constructor for **Parttime** class with invalid hourly rate format. The constructor requires the name, department, date hired, and hourly rate input data. Radio buttons from the GUI are used to collect certain data from the user.	**Instance #1:** “Smith, Joe”, CS radio button selected, “3/09/2021”, “Invalid hours” Part time radio button selected	The GUI rejects the attempt to add the employee and displays a message informing the user the hourly rate is an invalid format.
3	Test the parameterized constructor for **Parttime** class with a negative hourly rate. The constructor requires the name, department, date hired, and hourly rate input data. Radio buttons from the GUI are used to collect certain data from the user.	**Instance #1:** “Smith, Joe”, CS radio button selected, “3/09/2021”, “-37.50” Part time radio button selected	The GUI rejects the attempt to add the employee to the database and informs the user the hourly rate may not be a negative value.
4	Test the parameterized constructor for **Parttime** class with the absence of an hourly rate input at all. The constructor requires the name, department, date hired, and hourly rate input data. Radio buttons from the GUI are used to collect certain data from the user.	**Instance #1:** “Smith, Joe”, CS radio button selected, “3/09/2021” Part time radio button selected	The GUI rejects the attempt to add the employee to the database and informs the user they must enter an input for hourly rate.
5	Test the parameterized constructor for **Fulltime** class with all completely valid credentials. The constructor requires the name, department, date hired, and annual salary to instantiate an instance. Radio buttons from the GUI are used to collect	**Instance #1:** “Smith, Joe”, ECE radio button selected, “3/03/2021”, “85000”	The GUI prints the employee is successfully added to the database. The method for adding employees in the

	certain data from the user.	Full time radio button selected	company class returns true.
6	Test the parameterized constructor for **Fulltime** class with an invalid salary format. The constructor requires the name, department, date hired, and annual salary to instantiate an instance. Radio buttons from the GUI are used to collect certain data from the user.	**Instance #1:** “Smith, Joe”, ECE radio button selected, “3/03/2021”, “Salary” Full time radio button selected	The GUI rejects the attempt to add the employee to the database and informs the user the annual salary is an invalid format.
7	Test the parameterized constructor for **Fulltime** class with a negative salary amount. The constructor requires the name, department, date hired, and annual salary to instantiate an instance. Radio buttons from the GUI are used to collect certain data from the user.	**Instance #1:** “Smith, Joe”, ECE radio button selected, “3/03/2021”, “-85000”, Full time radio button selected	The GUI rejects the attempt to add the employee to the database and informs the user that the salary cannot be a negative value.
8	Test the parameterized constructor for **Management** class with all completely valid credentials. The constructor requires the name, department, date hired, annual salary, and manager role to instantiate an instance. Radio buttons from the GUI are used to collect certain data from the user.	**Instance #1:** “Smith, Joe”, IT radio button selected, “3/03/2021”, “90000”, Full time radio button selected, Director radio button selected	The GUI successfully displays a message informing the user that the employee was successfully added. The method for adding employees in company class returns true.
9	Test the parameterized constructor for **Management** class with a negative salary. The constructor requires the name, department, date hired, annual salary, and manager role to instantiate an instance. Radio buttons from the GUI are used to collect certain data from the user	**Instance #1:** “Smith, Joe”, IT radio button selected, “3/03/2021”, “-90000”, Full time radio button selected, Department Head radio button selected	The GUI rejects the attempt to add the employee to the database and informs the user that the salary cannot be a negative value.
10	Test the parameterized constructor for **Management** class with an invalid salary format. The constructor requires the name, department, date hired, annual salary, and manager role to instantiate an instance. Radio buttons from the GUI are	**Instance #1:** “Smith, Joe”, IT radio button selected, “3/03/2021”, “salary”,	The GUI rejects the attempt to add the employee to the database and informs the user the annual salary is an invalid

	used to collect certain data from the user	Full time radio button selected, Department Head radio button selected	format.
11	Test to set the hours with valid credentials for an existing **Parttime** employee.	**Instance #1:** “Smith, Joe”, CS radio button selected, “3/09/2021”, “37.50”, Part time radio button selected **Instance #2:** “Smith, Joe”, “3/09/2021”, “80” CS radio button selected, Part time radio button selected	The GUI successfully displays the working hours are set for the employee. The set hours method in company class returns true.
12	Test to set the hours with invalid credentials for an existing part time employee. The program expects an integer value for the hours to set field. The method for setting hours expects an integer but a string is provided instead.	**Instance #1:** “Smith, Joe”, CS radio button selected, “3/09/2021”, “37.50”, Part time radio button selected **Instance #2:** “Smith, Joe”, CS radio button selected, “3/09/2021”, Part time radio button selected, “hours”	The GUI rejects the attempt to set the hours for the employee and informs the user with a message that the hours worked must be an integer value.
13	Test to set the hours of an employee that does not exist. In this case the database is not empty however no employee in the	**Instance #1:** “Chris, Henry”, ECE radio button	The GUI rejects the attempt to set the hours and informs

	database should have their hours changed because the instance of the employee that the user wants to set hours for does not exist in the database.	selected, “3/03/2021”, “37.50”, Part time radio button selected **Instance #2:** “Smith, Joe”, CS radio button selected, “3/09/2021”, Part time radio button selected, “hours”	the user with a message that the employee does not exist in the database.
14	Test to remove an employee from an empty database. The parameters to remove an employee are the name, department, and date hired of the employee. The radio button is used to gather input for the department in the GUI.	**Instance #1:** “Smith, Joe”, CS radio button selected, “3/03/2019”	The GUI rejects the attempt to remove the employee from the database and informs the user that the database is empty.
15	Test to remove an employee that exists in the database. The parameters to remove an employee are the name, department, and date hired of the employee. The radio button is used to gather input for the department in the GUI.	**Instance #1:** “Smith, Joe”, CS radio button selected, “3/03/2019”, Full time radio button selected, “85000” **Instance #2:** “Smith, Joe”, CS radio button selected, “3/03/2019”	The GUI outputs a message displaying the employee was successfully removed from the database. The remove employee method in the company class returns true.
16	Test to add an employee to the database where the date hired is before the year 1900 which is the minimum accepted year for a date hired.	**Instance #1:** “Smith, Joe”, ECE radio button selected, “2/09/1899”, “37.50”, Part time radio button selected	The GUI rejects the attempt for the user to add the employee and displays a message saying the date is invalid.
17	Test to add an employee to the database where a name is not provided as an entry when submitting. For all types of	**Instance #1:** “2/09/2015”, ECE radio button	The GUI rejects the attempt to add the user to the database

	employees there must be a name provided to be successfully added to the database.	selected, “60000”, Fulltime radio button selected	and displays a message to provide the name of the employee.
18	Test to add an employee where no date is provided. For all types of employees at least a date must be provided for an employee to be successfully added to the database.	**Instance #1:** “Doe, Jane”, ECE radio button selected, “60000”, Management radio button selected, Manager radio button selected	The GUI rejects the attempt to add the user to the database and displays a message to choose a date for the date hired entry.
19	Test to print the current contents of the database to the text area on the GUI. The user will click on the print dropdown and select “All Employees” on the GUI.	**Instance #1:** “Smith, Joe”, “3/09/2021”, CS radio button selected, “37.50”, Part Time radio button selected **Instance #2:** “John, David”, “6/15/2017”, ECE radio button selected Full Time radio button selected, “85000” **Instance #3:** “Sara, McConnel”, “7/10/2019”, IT radio button selected Full Time radio button selected, “85000”	The GUI successfully outputs the current state of the database and employee attributes in the text area.
20	Test to print the contents of the database by employee department to the text area on the GUI. The user will click on the print	**Instance #1:** “Smith, Joe”, “3/09/2021”,	The GUI successfully outputs the employee attributes in the

	dropdown and select “By department” on the GUI.	CS radio button selected, “37.50”, Part Time radio button selected **Instance #2:** “John, David”, “6/15/2017”, ECE radio button selected Full Time radio button selected, “85000” **Instance #3:** “Sara, McConnel”, “7/10/2019”, IT radio button selected Full Time radio button selected, “85000” **Instance #4:** “Sara, McConnel”, “7/10/2019”, CS radio button selected Management radio button selected, “85000” Director radio button selected	database grouped by department in the text area.
21	Test to print the contents of employees in the database by date hired to the text area on the GUI. The user will click on the print dropdown and select “By date hired” on the GUI.	**Instance #1:** “Smith, Joe”, “3/09/2021”, CS radio button selected, “37.50”, Part Time radio button selected **Instance #2:** “John, David”, “6/15/2017”, ECE radio button selected Full Time radio button selected, “85000”	The GUI successfully outputs the employee attributes in the database by date hired in ascending order in the text area.

		Instance #3: “Sara, McConnel”, “7/10/2019”, IT radio button selected Full Time radio button selected, “85000”	
22	Test to print the contents of an empty database by selecting “All Employees” in the print dropdown in the GUI.	N/A	The GUI will reject the attempt to print the empty database and inform the user that the database is empty.
23	Test to import a text file to the database and populate the employees in the file to the current database. The employees should be added to the database regardless of whether the database is currently empty or not. The user imports the database by selecting the “File” -> “Import” menu item.	**Instance #1:** “Smith, Joe”, “3/09/2021”, CS radio button selected, “37.50”, Part Time radio button selected **Instance #2:** A database.txt file containing valid add employee commands.	The GUI will display an import success message if the file was of proper format and the file contained employees to be added. The database now contains the new employees as well as the employees that existed before the import.
24	Test to import a text file to the database in which the contents of the text file are not valid for importing employees.	A database.txt file containing invalid add employee commands in incorrect format.	The GUI will display a message saying there was an error in trying to import employees to the database and the current database is left unaffected by the attempt.
25	Test to export the contents of the current database to a text file to the location of the user’s choosing on the PC. The user exports the database by selecting the “File” -> “Export” menu item.	**Instance #1:** “Smith, Joe”, “3/09/2021”, CS radio button selected, “37.50”, Part Time radio button selected **Instance #2:** “John, David”,	The GUI will display a message saying the export was a success. The text file will contain the attributes of employees that exist in the database at the time of the export. No changes to the

		“6/15/2017”, ECE radio button selected Full Time radio button selected, “85000” **Instance #3:** “Sara, McConnel”, “7/10/2019”, IT radio button selected Full Time radio button selected, “85000”	database are made.
26	Test to export the contents of the database to a text file when the database is empty. The user exports the database by selecting the “File” -> “Export” menu item.	**N/A**	The GUI rejects the attempt to export the database and informs the user that the database is empty.
27	Test to compute the payments of the employees in the database. The user computes the payments by selecting the “Payment” -> “Process Payments” menu item.	**Instance #1:** “Smith, Joe”, “3/09/2021”, CS radio button selected, “37.50”, Part Time radio button selected **Instance #2:** “John, David”, “6/15/2017”, ECE radio button selected Full Time radio button selected, “85000” **Instance #3:** “Sara, McConnel”, “7/10/2019”, IT radio button selected Full Time radio button selected, “85000”	The GUI will display a message informing the user the payments were successfully computed. When the user next prints the contents of the database they will be able to see the payments to be made to each employee in the database.
28	Test to compute the payments of an empty database. The user computes the payments by selecting the “Payment” ->	**N/A**	The GUI rejects the attempt to compute the payments and

	“Process Payments” menu item.		informs the user that the database is empty.