|  |  |
| --- | --- |
| **Test Case No.** | **TXT-001** |
| **Title** | **Create a class object with attributes** |
| **Description** | **Create a class object with public attributes** |
| **Precondition** | **N/A** |
| **Assumption** | **N/A** |
| **Test Steps** | **Input the following data in the text area:**  **class Employee**  **+ date\_of\_join : date**  **+ eng\_name : string**  **+ chi\_name : string** |
| **Expected Result** | **A UML\_Class object named “Employee” is created with the following attributes:**   |  |  |  | | --- | --- | --- | | **public** | **date\_of\_join** | **date** | | **public** | **eng\_name** | **string** | | **public** | **chi\_name** | **string** | |

|  |  |
| --- | --- |
| **Test Case No.** | **TXT-002** |
| **Title** | **Create a class object with attributes of different modifiers** |
| **Description** | **Create a class object with private, protected, public and inner attributes** |
| **Precondition** | **N/A** |
| **Assumption** | **N/A** |
| **Test Steps** | **Input the following data in the text area:**  **class Employee**  **- date\_of\_join : date**  **+ eng\_name : string**  **+ chi\_name : string**  **# has\_dependent: bool**  **~ hkid\_no: string** |
| **Expected Result** | **A UML\_Class object named “Employee” is created with the following attributes:**   |  |  |  | | --- | --- | --- | | **private** | **date\_of\_join** | **date** | | **public** | **eng\_name** | **string** | | **public** | **chi\_name** | **string** | | **protected** | **has\_dependent** | **boolean** | | **package** | **hkid\_no** | **String** | |

|  |  |
| --- | --- |
| **Test Case No.** | **TXT-003** |
| **Title** | **Create a sub-class object** |
| **Description** | **Create a class object which extends another class** |
| **Precondition** | **N/A** |
| **Assumption** | **N/A** |
| **Test Steps** | **Input the following data in the text area:**  **class Employee**  **- date\_of\_join : date**  **- eng\_name : string**  **- chi\_name : string**  **- has\_dependent: bool**  **- hkid\_no: string**  **Class Worker >> Employee** |
| **Expected Result** | **(1) A UML\_Class object named “Employee” is created with the following attributes:**   |  |  |  | | --- | --- | --- | | **private** | **date\_of\_join** | **date** | | **private** | **eng\_name** | **string** | | **private** | **chi\_name** | **string** | | **private** | **has\_dependent** | **boolean** | | **private** | **hkid\_no** | **String** |   **(2) A UML\_Class object named “Worker” is created without any attributes. The class has a parent named “Employee”.** |

|  |  |
| --- | --- |
| **Test Case No.** | **TXT-004** |
| **Title** | **Create a sub-class object** |
| **Description** | **Create a class object which extends another class which does not exist** |
| **Precondition** | **N/A** |
| **Assumption** | **N/A** |
| **Test Steps** | **Input the following data in the text area:**  **Class Worker >> Employee** |
| **Expected Result** | **A UML\_Class object named “Employee” is created without any attributes or methods. NULL value is stored in its parent.** |

|  |  |
| --- | --- |
| **Test Case No.** | **TXT-005** |
| **Title** | **Create a class object with methods** |
| **Description** | **Create a class object with methods** |
| **Precondition** | **N/A** |
| **Assumption** | **N/A** |
| **Test Steps** | **Input the following data in the text area:**  **class Employee**  **- date\_of\_join : date**  **- eng\_name : string**  **- chi\_name : string**  **- salary : double**  **- hkid\_no: string**  **+ increaseSalary(double percentage): void**  **+ getEmployeeFullName(): string** |
| **Expected Result** | **A UML\_Class object named “Employee” is created with the following attributes and methods:**   |  |  |  | | --- | --- | --- | | **private** | **date\_of\_join** | **date** | | **private** | **eng\_name** | **string** | | **private** | **chi\_name** | **string** | | **private** | **salary** | **double** | | **private** | **hkid\_no** | **String** |  |  |  |  | | --- | --- | --- | | **public** | **increaseSalary() parameters:**  **percentage (double)** | **void** | | **public** | **getEmployeeFullName()** | **String** | | **private** | **hasWorkingVisa()** | **boolean** | |

|  |  |
| --- | --- |
| **Test Case No.** | **TXT-006** |
| **Title** | **Create an interface object with methods** |
| **Description** | **Create an interface object with methods** |
| **Precondition** | **N/A** |
| **Assumption** | **N/A** |
| **Test Steps** | **Input the following data in the text area:**  **interface IExpatriate**  **+ applyWorkingVisa (date expirydate): boolean**  **+ claimExpense(date claim\_date, double claim\_amount): boolean** |
| **Expected Result** | **An UML\_Interface object named “IExpatriate” is created with the following methods:**   |  |  |  | | --- | --- | --- | | **public** | **applyWorkingVisa() parameters:**  **expirydate (date)** | **boolean** | | **public** | **claimExpense ()**  **parameters:**  **claim\_date (date)**  **claim\_amoun (double)** | **boolean** | |

|  |  |
| --- | --- |
| **Test Case No.** | **TXT-007** |
| **Title** | **Create an interface object with methods** |
| **Description** | **Create an interface object with methods** |
| **Precondition** | **N/A** |
| **Assumption** | **N/A** |
| **Test Steps** | **Input the following data in the text area:**  **interface IExpatriate**  **+ applyWorkingVisa (date expirydate): boolean**  **+ claimExpense(date claim\_date, double claim\_amount): boolean**  **class Employee || IExpatriate**  **- date\_of\_join : date**  **- eng\_name : string**  **- chi\_name : string**  **- salary : double**  **- hkid\_no: string**  **+ increaseSalary(double percentage): void**  **+ getEmployeeFullName(): string** |
| **Expected Result** | **A UML\_Class object named “Employee” is created with the following attributes and methods:**   |  |  |  | | --- | --- | --- | | **private** | **date\_of\_join** | **date** | | **private** | **eng\_name** | **string** | | **private** | **chi\_name** | **string** | | **private** | **salary** | **double** | | **private** | **hkid\_no** | **String** |  |  |  |  | | --- | --- | --- | | **public** | **increaseSalary() parameters:**  **percentage (double)** | **void** | | **public** | **getEmployeeFullName()** | **String** |   **A UML\_Interface object named “IExpatriate” is created with the following methods:**   |  |  |  | | --- | --- | --- | | **public** | **applyWorkingVisa() parameters:**  **expirydate (date)** | **boolean** | | **public** | **claimExpense ()**  **parameters:**  **claim\_date (date)**  **claim\_amoun (double)** | **boolean** |   **The UML\_Class object Employee implements the UML\_Interface object IExpatriate.** |

|  |  |
| --- | --- |
| **Test Case No.** | **TXT-008** |
| **Title** | **Create an interface object with methods** |
| **Description** | **Create an interface object with methods** |
| **Precondition** | **N/A** |
| **Assumption** | **N/A** |
| **Test Steps** | **Input the following data in the text area:**  **interface IExpatriate**  **+ applyWorkingVisa (date expirydate): boolean**  **+ claimExpense(date claim\_date, double claim\_amount): boolean**  **interface ITravelBenefit**  **+ assignDriver (string driver\_name): void**  **+ claimTravelExpense(date claim\_date, double claim\_amount): boolean**  **class Employee || IExpatriate || ITravelBenefit**  **- date\_of\_join : date**  **- eng\_name : string**  **- chi\_name : string**  **- salary : double**  **- hkid\_no: string**  **+ increaseSalary(double percentage): void**  **+ getEmployeeFullName(): string** |
| **Expected Result** | **A UML\_Class object named “Employee” is created with the following attributes and methods:**   |  |  |  | | --- | --- | --- | | **private** | **date\_of\_join** | **date** | | **private** | **eng\_name** | **string** | | **private** | **chi\_name** | **string** | | **private** | **salary** | **double** | | **private** | **hkid\_no** | **String** |  |  |  |  | | --- | --- | --- | | **public** | **increaseSalary() parameters:**  **percentage (double)** | **void** | | **public** | **getEmployeeFullName()** | **String** |   **A UML\_Interface object named “IExpatriate” is created with the following methods:**   |  |  |  | | --- | --- | --- | | **public** | **applyWorkingVisa() parameters:**  **expirydate (date)** | **boolean** | | **public** | **claimExpense ()**  **parameters:**  **claim\_date (date)**  **claim\_amount (double)** | **boolean** |   **A UML\_Interface object named “ITravelBenefit” is created with the following methods:**   |  |  |  | | --- | --- | --- | | **public** | **assignDriver() parameters:**  **driver\_name (string)** | **void** | | **public** | **claimTravelExpense ()**  **parameters:**  **claim\_date (date)**  **claim\_amount (double)** | **boolean** |   **The UML\_Class object Employee implements the UML\_Interface objects IExpatriate and ITravelBenefit.** |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 1 - validate Interface Association Line |
| Description | Test normal input to validate Interface Association Line |
| Precondition | Input normal text into text area |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate >> ILicenseTest |
| Expected Result | Return True result |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 2 - validate Interface Association Line |
| Description | Test input only have one > which separated by a name in between |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate > ILicenseTest |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 3 - validate Interface Association Line |
| Description | Test no “>” with the string |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate ILicenseTest |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 4 - validate Interface Association Line |
| Description | Test input more than two “>” in the string |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate >>> ILicenseTest |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 5 - validate Interface Association Line |
| Description | Test input “>>” at the start of the string |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  >> interface IExpatriate ILicenseTest |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 6 - validate Interface Association Line |
| Description | Test input “>>” at the end of the string |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate ILicenseTest >> |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 1 – Validate Class Association Line |
| Description | Test normal input to validate Class Association Line |
| Precondition | Input normal text into text area |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  **class Employee || IExpatriate** |
| Expected Result | Return True result |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 2 – Validate Class Association Line |
| Description | Test input only have one > which separated by a name in between |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate > ILicenseTest |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 3 - validate Class Association Line |
| Description | Test no “>” with the string |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate ILicenseTest |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 4 - validate Class Association Line |
| Description | Test input more than two “>” in the string |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate >>> ILicenseTest |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 5 - validate Class Association Line |
| Description | Test input “>>” at the start of the string |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  >> interface IExpatriate ILicenseTest |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title | Test Case 6 - validate Class Association Line |
| Description | Test input “>>” at the end of the string |
| Precondition | N/A |
| Assumption | N/A |
| Test Steps | Input the following data in the text area  interface IExpatriate ILicenseTest >> |
| Expected Result | Return False result and prompt error message to user |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title |  |
| Description |  |
| Precondition |  |
| Assumption |  |
| Test Steps |  |
| Expected Result |  |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title |  |
| Description |  |
| Precondition |  |
| Assumption |  |
| Test Steps |  |
| Expected Result |  |

|  |  |
| --- | --- |
| Test Case No. |  |
| Title |  |
| Description |  |
| Precondition |  |
| Assumption |  |
| Test Steps |  |
| Expected Result |  |