**Test Cases and Test Descriptions**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Delivery | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| Delivery | 1 | Num items is greater than 0 | Delivery(Reg, MML, 3, true) | reciepientName = Reg  Region = MML  numItems = 3  isInsured = true | reciepientName = Reg  Region = MML  numItems = 3  isInsured = true | P |
| 2 | Num items is less than or equal to 0 | Delivery(Reg, MML, 0, true) | reciepientName = Reg  Region = MML  numItems = 1  isInsured = true  (Although with GUI, user is prompted to enter again) | reciepientName = Reg  Region = MML  numItems = 1  isInsured = true  (Although with GUI, user is prompted to enter again) | P |
| 3 | Items not insured | Delivery(Reg, MML, 3, false) | reciepientName = Reg  Region = MML  numItems = 1  isInsured = false | reciepientName = Reg  Region = MML  numItems = 1  isInsured = false | P |
| computeTotalFee | 1 | Chosen destination is MNL.  Parcel is FLT 2.  Parcel is Insured. | Parcel = Large Flat  isInsured = True | 105 | 105 | P |
| 2 | Chosen destination is MIN.  Parcel’s computed weight cost is less than 3000.  Parcel is not Insured. | Parcel’s Weight = 1000 grams  isInsured = False | 3040 | 3040 | P |
| 3 | Chosen destination is MIN.  Parcel’s computed weight cost is greater than 3000.  Parcel is not Insured. | Parcel’s Weight = 20000 grams  isInsured = False | 5040 | 5040 | P |
| dDateAdjust | 1 | Entered number of days is greater than or equal to 0 | K = 2 | date today= 809  delivery date =811 | date today= 809  delivery date =811 | P |
| 2 | Entered number of days is less than 0 | K = -1 | Display “Error” in cmd only | Display “Error” in cmd only | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DeliveryInfos | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| addDelivery | 1 | Delivery is valid | D is a correct delivery | D is added to ArrayList of Deliveries | D is added to ArrayList of Deliveries | P |
| UpdateDelivery | 1 | ArrayList of Deliveries is not empty. | ArrayList has 2 deliveries | Update status of 2 deliveries | Update status of 2 deliveries | P |
| 2 | ArrayList of Deliveries is empty | ArrayList has no deliveries | Does Nothing | Did Nothing | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Dimension | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| Dimension | 1 | Created Dimension has length, width, height, and weight component | Dimension(3,3,4,3) | Length = 3  Width = 3  Height = 4  Weight = 3 | Length = 3  Width = 3  Height = 4  Weight = 3 | P |
| 2 | Created Dimension has length, width, height components only | Dimension(3,3,4) | Length = 3  Width = 3  Height = 4  Weight = 0 | Length = 3  Width = 3  Height = 4  Weight = 0 | P |
| addLength | 1 | L is greater than or equal to zero | L = 3  Length = 0 | Length = 3 | Length = 3 | P |
| addWidth | 1 | W is greater than or equal to zero | W= 3  Width = 0 | Width = 3 | Width = 3 | P |
| addHeight | 1 | h is greater than or equal to zero | h = 3  Height = 0 | Height = 3 | Height = 3 | P |
| addWidth | 1 | W is greater than or equal to zero | W= 3  Weight = 0 | Weight = 3 | Weight = 3 | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Document | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| Document | 1 | Created Document has name, length, width, pages | Document(docu, 8.5,11,25) | Name: docu  Length = 8.5  Width = 11  Height = 1  Weight = 1  Pages = 25 | Name: docu  Length = 8.5  Width = 11  Height = 1  Weight = 1  Pages = 25 | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| IrregularObj | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| IrregularObj | 1 | Created Irregular Object has name, length, width, height, weight | IrregularObj(rocks, 8.5,11,2,2) | Name: rocks  Length = 8.5  Width = 11  Height = 2  Weight = 2 | Name: Rocks  Length = 8.5  Width = 11  Height = 2  Weight = 2 | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| JMController | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| actionPerformed  (“Make a Shipment”) | 1 | Chosen by user in the menu | Interaction | Takes user to another window and is prompted to enter recipient,  Shipping destination, Number of Items and insurance coverage | Takes user to another window and is prompted to enter recipient,  Shipping destination, Number of Items and insurance coverage | P |
| actionPerformed  (“Track an Item”) | 1 | Chosen by user in the menu | Interaction | Takes user to another window and is prompted to enter Shipping Number to track his/her shipping | Takes user to another window and is prompted to enter Shipping Number to track his/her shipping | P |
| actionPerformed  (“Search”) | 1.1 | Shipping Exist | Shipping number | Display Shipping Details | Displayed Shipping Details | P |
|  | 1.2 | Shipping Does Not Exist | Shipping number | Display Error Message “No Results Found!” | Displayed Error Message “No Results Found!” | P |
| actionPerformed  (“Admin Menu”) | 1 | Chosen by user in the menu | Interaction | Prompts user to enter admin password to be able to enter menu | Prompts user to enter admin password to be able to enter menu | P |
| 1.1 | Password is Wrong | Interaction | User is taken back to Main Menu | User is taken back to Main Menu | P |
| 1.2 | Password is Right | Interaction | User is taken to Admin Menu Screen | User is taken to Admin Menu Screen | P |
| actionPerformed  (“Basic Report”) | 1 | Chosen by user in the menu | Interaction | Shows user list of pending/processing shipments | Shows user list of pending/processing shipments | P |
| actionPerformed  (“Advance Report”) | 1 | Chosen by user in the menu | Interaction | Shows user list of all shipments | Shows user list of all shipments | P |
| actionPerformed  (“Edit Item Details”) | 1 | All fields in the Shipping Menu are filled | All textboxes are filled, cmbDest’s index is greater than 0 | Will proceed to the Edit Item Details Menu | Proceeded to the Edit Item Details Menu | P |
|  | 2 | Some fields in the Shipping Menu are empty/entered a wrong entry type | tfRecipient is empty | Display Error Message “Please fill up all items correctly!” | Error Message “Please fill up all items correctly!” displayed on GUI | P |
| actionPerformed  (“Check Available Parcels”) | 1 | All fields are answered correctly | All textboxes are filled and answered correctly | Shows User list of Possible parcels via drop down and allows user to select | Shows User list of Possible parcels via drop down and allows user to select | P |
|  | 2 | Not all fields are answered correctly | Not all textboxes are filled and answered correctly | Display Error Message “Please fill up all items correctly!” | Display Error Message “Please fill up all items correctly!” | P |
| checkAvailableParcels | 1 | There exists a suitable parcel for the items | Items fit in a parcel | Sets the list of possible parcels | Sets the list of possible parcels | P |
|  | 2 | There is no suitable parcel for the items | Items do not fit in any parcel | Possible Parcels list is empty | Possible Parcels List is empty | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| JMGUI | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| adjustDate | 1 | Updates date | Updates date | Date is updated | Date is updated | P |
| checkEditItemsFull | 1 | Edit Item is Full | All items are filled | Return true | Returned true | P |
| 2 | Items are not filled | Not all fields are filled up complete and correctly | Return false | Returned false | P |
| checkAddItemsFull | 1 | Items are filled | All items are filled | Return true | Returned true | P |
| 2 | Items are not filled | Not all fields are filled up complete and correctly | Return false | Returned false | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Item | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| Item | 1 | Created Item name, length, width, height, weight | Item (object, 8.5,11,2,2) | Name: object  Length = 8.5  Width = 11  Height = 2  Weight = 2 | Name: object  Length = 8.5  Width = 11  Height = 2  Weight = 2 | P |
| WeightComparator  (compare) | 1 | Item 1 is heavier than item 2 | Item 1 = 3kg  Item 2 = 1kg | Returns -1 | Returns -1 | P |
|  | 2 | Item 2 is heavier than item 1 | Item 1 = 1kg  Item 2 = 3kg | Returns 1 | Returns 1 | P |
|  | 3 | Item 2 is of equal weight with item 1 | Item 1 = 1kg  Item 2 = 1kg | Returns 0 | Returns 0 | P |
| LengthComparator  (compare) | 1 | Item 1 is longer than item 2 | Item 1 = 3”  Item 2 = 1” | Returns -1 | Returns -1 | P |
|  | 2 | Item 2 is longer than item 1 | Item 1 = 1”  Item 2 = 3” | Returns 1 | Returns 1 | P |
|  | 3 | Item 2 is of equal length with item 1 | Item 1 = 1”  Item 2 = 1” | Returns 0 | Returns 0 | P |
| WidthComparator  (compare) | 1 | Item 1 is wider than item 2 | Item 1 = 3”  Item 2 = 1” | Returns -1 | Returns -1 | P |
|  | 2 | Item 2 is wider than item 1 | Item 1 = 1”  Item 2 = 3” | Returns 1 | Returns 1 | P |
|  | 3 | Item 2 is of equal width with item 1 | Item 1 = 1”  Item 2 = 1” | Returns 0 | Returns 0 | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Parcel | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| Parcel | 1 | Total item count is greater than 0 | Parcel(2) | TotItemCount = 2 | TotItemCount = 2 | P |
| removePossibleParcels | 1 | Possibleparcels has content | removePossibleParcels(0) | Element 0 is remove | Element 0 is remove | P |
| 2 | Possibleparcels has no content | removePossibleParcels(0) | Nothing happens  GUI displays No Possible Parcels | Nothing happens  GUI displays No Possible Parcels | P |
| addItem | 1 | Item is document | Item(d1) | Item of subtype Document is added to arraylist of items | Item of subtype Document is added to arraylist of items | P |
| 2 | Item is an irregular object | Item(irregobj1) | Item of subtype Irregular object is added to arraylist of items | Item of subtype Irregular Object is added to arraylist of items | P |
| 3 | Item is a Regular object | Item(regobj1) | Item of subtype Regular object is added to arraylist of items | Item of subtype Regular Object is added to arraylist of items | P |
| sortItemDimension | 1 | Item length is the biggest dimension followed by width and height | L = 11, w=7 h= 2 | L = 11, w=7 h= 2 | L = 11, w=7 h= 2 | P |
| 2 | Item width is the biggest dimension followed by length and height | L = 7, w=11 h= 2 | L = 11, w=7 h= 2 | L = 11, w=7 h= 2 | P |
| 3 | Item height is the biggest dimension followed by width and length | L = 2, w=7 h= 11 | L = 11, w=7 h= 2 | L = 11, w=7 h= 2 | P |
| doesFit | 1 | Item fits in dimensions of parcel | Item : 8x11x1  Parcel 9x14x1 | Dimension is added to list of possible parcel dimension | Dimension is added to list of possible parcel dimension | P |
| 2 | Item does not fit in dimensions of parcel | Item : 9x11x1  Parcel 9x14x1 | Dimension is not added to list of possible parcel dimension | Dimension is not added to list of possible parcel dimension | P |
| 3 | No parcel fits the item | All parcels are not able to encapsulate item | Possible Parcel dimension list is empty | Possible Parcel Dimension list is empty | P |
| checkAvailableParcels | 1 | Total weight is over 3 kilograms | Totweight = 4kg | Checks for possible parcel sizes based on LxWxH but removes first and second parcel if included in the list | Checks for possible parcel sizes based on LxWxH but removes first and second parcel if included in the list | P |
| 2 | Total weight is below or equal to 3 kilograms | Total Weight= 2kg | Checks for possible parcel sizes based on LxWxH | Checks for possible parcel sizes based on LxWxH | P |
| displayPossibleParcels | 1 | PossibleParcels is not empty | PossibleParcels has content | Display all content of PossibleParcels | Display all content of PossibleParcels | P |
| 2 | PossibleParcels is empty | PossibleParcels has no content | Display “No Available Parcels” | Display “No Available Parcels” | P |
| getTotalWeightofIrregObj | 1 | There is an Irregular object in the Parcel | Parcel has Irregular object | Returns sum of weight of irregular objects | Returns sum of weight of irregular objects | P |
| 2 | There is no Irregular object in the Parcel | Parcel has no Irregular object | Returns 0 | Returns 0 | P |
| getTotalVolWeightofIrregObj | 1 | There is an Irregular object in the Parcel | Parcel has Irregular object | Returns sum of Volumetric weight of irregular objects | Returns sum of Volumetric weight of irregular objects | P |
| 2 | There is no Irregular object in the Parcel | Parcel has no Irregular object | Returns 0 | Returns 0 | P |
| getTotalWeightofNonIrregObj | 1 | There is a Regular object/Document in the Parcel | Parcel has Regular Object/Document | Returns sum of weight of Regular Objects/Documents | Returns sum of weight of Regular Objects/Documents | P |
| 2 | There is a Regular object/Document in the Parcel | Parcel has no Regular Object/Document | Returns 0 | Returns 0 | P |
| computeParcelCost | 1 | Parcel Choice is small flat pouch (0) | 0 | 30 + IrregObj VolWeight/Weight computation | 30 + IrregObj VolWeight/Weight computation | P |
| 2 | Parcel Choice is large flat pouch (1) | 1 | 50 + IrregObj VolWeight/Weight computation | 50 + IrregObj VolWeight/Weight computation | P |
| 3 | Parcel Choice is box (2-5) | 3 | 40\*weight of nonIrregObj + IrregObj Volweight/  WeightComputation | 40\*weight of nonIrregObj + IrregObj Volweight/  WeightComputation | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Product | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| Product | 1 | Created Product has name, length, width, height, weight | Product(rocks, 8.5,11,2,2) | Name: rocks  Length = 8.5  Width = 11  Height = 2  Weight = 2 | Name: Rocks  Length = 8.5  Width = 11  Height = 2  Weight = 2 | P |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tracker | | | | | | |
| Method | # | Test Description | Sample Input | Expected Output | Actual Output | P/F |
| showSpecificShipping | 1 | Tracking input is found within the Tracker object class. | FLT0701MNL01001  (Assume that it exists in the class) | Will display the status of the Delivery with matching tracking # | Displayed the status of the Delivery with matching tracking # | P |
|  | 2 | Input is not found within the Tracker object class. | FLT0702MNL01003  (Assume that it does not exist in the class) | Display Error Message “No Results Found” | Error Message “No Resullts Found” displayed on GUI | P |