**Algorithm:**

*# This function for ordering ticket Return or One-Way***def** ordering():  
 *""" The main Function """* passenger = choosePassenger(username)  
 ticketSelection = ticketType()  
 **if** ticketSelection == **"Return"**:  
 flight = ticketReturn()  
 **elif** ticketSelection == **"One-Way"**:  
 flight = ticketOne()  
 classChoice = chooseClass()  
 seatType = chooseSeat()  
 age = ageInput()  
 price = countTotal(ticketSelection, flight, classChoice, seatType) *# Count the total fare* print(**"Calculating fare . . . "**)  
 print(**"Ticket for: "** + passenger)  
 print(ticketSelection)  
 print(flight)  
 print(classChoice)  
 print(seatType)  
 discountedPrice = discountCheck(age, price) *# Giving discount for people < 16 year old* print(**"Total Price: ${}"**.format(discountedPrice))  
 priceList.append(discountedPrice) *# priceList to listed all people's order, so the total could be counted in the (E)xit option* itemList.append(**"${:.2f}"**.format(discountedPrice))

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
|  | Call function choosePassenger to select their ticket whether Return or One-Way ticket  Call function ticket type  If ticket selection = return  Call function ticketReturn  If ticketSelection = One-Way  Call function ticketOne  Call function chooseClass  Call function chooseSeat  Get age form the user  Call function countTotal (ticketSelection, flight, classChoice,seatType)  Call function discountCheck (age, price)  Using List Append  pricelist.append (discountedPrice)  item.List.append (“${:.2f}”.format(discountedPrice)) | Print (“Calculating fare . . .”)  Print (“Ticket for: “, passenger)  Print (ticketSelection)  Print (classChoice)  Print (seatType)  Print (“Total Price: ${}”. Format (discountedPrice)) |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| Return  One-Way | ticketSelection == **"Return"**  ticketSelection == **"One-Way"** | Call function ticketReturn  Call function ticketOne |

**Error Checking Loop**

Loop variable – ticketSelection, ticketReturn, ticketOne

Exit Condition – ticketSelection == **"Return" or** ticketSelection == **"One-Way"**

**Algorithm**

*# This is Main Menu function for giving Instruction, Order, and Exit***def** mainMenu():  
 print(**"Tropical Airlines Ticket Ordering System"**)  
 print(**"(I)nstructions"**)  
 print(**"(O)rder ticket"**)  
 print(**"(E)xit"**)  
 menuSelection = input()  
 menuSelection = menuSelection.upper()  
 **return** menuSelection

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Ouput** |
|  | Get menuSelection from user | Return menuSelection  print(**"Tropical Airlines Ticket Ordering System"**) print(**"(I)nstructions"**) print(**"(O)rder ticket"**) print(**"(E)xit"**) |

**Algorithm**

*# This Function to order ticker for yourself or someone else***def** choosePassenger(username):  
 **while True**:  
 print(username, **" is this ticket for:"**)  
 print(**"(Y)ou"**)  
 print(**"(S)omeone else"**)  
 menuChoice = input()  
 menuChoice = menuChoice.upper()  
 **if** menuChoice == **"Y"**:  
 passenger = username  
 **break  
 elif** menuChoice == **"S"**:  
 passenger = input(**"Please enter the name of the person travelling"**)  
 **break  
 else**:  
 print(**"Invalid menu Choice"**)  
 **return** passenger

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
| Username | Get menuChoice form user | Return passenger  print(username, **" is this ticket for:"**) print(**"(Y)ou"**)  print(**"(S)omeone else"**)  print(**"Invalid menu Choice"**) |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| This ticket for ..  Yourself  Someone else | While = True  menuChoice == **"Y"**  menuChoice == **"S"** | print(username, **" is this ticket for:"**) print(**"(Y)ou"**) print(**"(S)omeone else"**) menuChoice = input() menuChoice = menuChoice.upper()Call function ticketOne  passenger = username  break  get passenger from user |

**Error Checking Loop**

Loop variable – menuChoice

Exit Condition – menuChoice == **"Y"** **or** menuChoice == **"S"**

**Algorithm**

*# This function for giving options to select ticket Return or On-Way ticket***def** ticketType():  
 **while True**:  
 print(**"Is this a"**)  
 print(**"(R)eturn trip"**)  
 print(**"(O)ne-Way"**)  
 ticketSelection = input()  
 ticketSelection = ticketSelection.upper()  
 **if** ticketSelection == **"R"**:  
 typeChoice = **"Return"  
 break  
 elif** ticketSelection == **"O"**:  
 typeChoice = **"One-Way"  
 break  
 else**:  
 print(**"Invalid menu Choice"**)  
 **return** typeChoice

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
|  | Get ticketSelection from user | Return typeChoice  print(**"Is this a"**) print(**"(R)eturn trip"**) print(**"(O)ne-Way"**)  print(**"Invalid menu Choice"**) |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| Choose Return ticket or One-Way ticket  Return  One-Ways | While = True  ticketSelection == **"R"**  ticketSelection == **"O"** | print(**"Is this a"**) print(**"(R)eturn trip"**) print(**"(O)ne-Way"**) ticketSelection = input() ticketSelection = ticketSelection.upper()  typeChoice = **"Return" break**  typeChoice = **"Return" break**  typeChoice = **"One-Way" break** |

**Error Checking Loop**

Loop variable – ticketSelection, ticketChoice

Exit Condition – ticketSelection == **"Return" or** ticketSelection == **"One-Way"**

**Algorithm**

*# This function to run the Return ticket function***def** ticketReturn():  
 **while True**:  
 print(**"Please select the destination for your return trip. Fare prices are listed below."**)  
 print(**"(C)airns - $400"**)  
 print(**"(S)ydney - $575"**)  
 print(**"(P)erth - $700"**)  
 ticketTypes = input()  
 ticketTypes = ticketTypes.upper()  
 **if** ticketTypes == **"C"**:  
 flight = **"Cairns"  
 break  
 elif** ticketTypes == **"S"**:  
 flight = **"Sydney"  
 break  
 elif** ticketTypes == **"P"**:  
 flight = **"Perth"  
 break  
 else**:  
 print(**"Invalid menu Choice"**)  
 **return** flight

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
|  | Get ticketTypes from the user | Return flight  print(**"Please select the destination for your return trip. Fare prices are listed below."**) print(**"(C)airns - $400"**) print(**"(S)ydney - $575"**) print(**"(P)erth - $700"**)  print(**"Invalid menu Choice"**) |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| Choose Return ticket (C)airns, (S)ydney, and (P)erth  Cairns  Sydney  Perth | While = True  ticketTypes == **"C"**  ticketTypes == **"S"**  ticketTypes == **"P"** | print(**"Please select the destination for your return trip. Fare prices are listed below."**) print(**"(C)airns - $400"**) print(**"(S)ydney - $575"**) print(**"(P)erth - $700"**) ticketTypes = input() ticketTypes = ticketTypes.upper()  flight = **"Cairns" break**  flight = **"Sydney" break**  flight = **"Perth" break** |

**Error Checking Loop**

Loop variable – ticketTypes

Exit Condition – ticketTypes == **"C"** **or** ticketTypes == **"S" or** ticketTypes == **"P"**

**Algorithm**

*# This function is to run the One-Way ticket function***def** ticketOne():  
 **while True**:  
 print(**"Please select the destination for your One-Way trip. Fare prices are listed below."**)  
 print(**"(C)airns - $250"**)  
 print(**"(S)ydney - $420"**)  
 print(**"(P)erth - $510"**)  
 ticketSelecetions = input()  
 ticketSelecetions = ticketSelecetions.upper()  
 **if** ticketSelecetions == **"C"**:  
 flight = **"Cairns"  
 break  
 elif** ticketSelecetions == **"S"**:  
 flight = **"Sydney"  
 break  
 elif** ticketSelecetions == **"P"**:  
 flight = **"Perth"  
 break  
 else**:  
 print(**"Invalid menu Choice"**)  
 **return** flight

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
|  | Get ticketSelections from user | Return flight |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| Choose One-Way ticket (C)airns, (S)ydney, and (P)erth  Cairns  Sydney  Perth | While = True  ticketTypes == **"C"**  ticketTypes == **"S"**  ticketTypes == **"P"** | print(**"Please select the destination for your One-Way trip. Fare prices are listed below."**) print(**"(C)airns - $250"**) print(**"(S)ydney - $420"**) print(**"(P)erth - $510"**) ticketSelecetions = input() ticketSelecetions = ticketSelecetions.upper()  flight = **"Cairns" break**  flight = **"Sydney" break**  flight = **"Perth" break** |

**Error Checking Loop**

Loop variable – ticketTypes

Exit Condition – ticketTypes == **"C"** **or** ticketTypes == **"S" or** ticketTypes == **"P"**

**Algorithm**

*# This function for Choosing type of class fare***def** chooseClass():  
 **while True**:  
 print(**"Please choose the type of fare. Fees are displayed below and are in addition to the basic fare. Please note choosing Frugal fare means you will not be offered a seat choice, it will be assigned to the ticketholder at travel time."**)  
 print(**"(B)usiness – $275"**)  
 print(**"(E)conomy – $25"**)  
 print(**"(F)rugal – $0"**)  
 classSelection = input()  
 classSelection = classSelection.upper()  
 **if** classSelection == **"B"**:  
 classChoice = **"Business"  
 break  
 elif** classSelection == **"E"**:  
 classChoice = **"Economy"  
 break  
 elif** classSelection == **"F"**:  
 classChoice = **"Frugal"  
 break  
 else**:  
 print(**"Invalid menu choice"**)  
 **return** classChoice

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
|  | Get classSelection from user | Return classChoice |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| Choose One-Way ticket (C)airns, (S)ydney, and (P)erth  Business  Economy  Frugal | While = True  classSelection == **"B"**  classSelection == **"E"**  classSelection == **"F"** | print(**"Please choose the type of fare. Fees are displayed below and are in addition to the basic fare. Please note choosing Frugal fare means you will not be offered a seat choice, it will be assigned to the ticketholder at travel time."**) print(**"(B)usiness – $275"**) print(**"(E)conomy – $25"**) print(**"(F)rugal – $0"**) classSelection = input() classSelection = classSelection.upper()  classChoice = **"Business" break**  classChoice = **"Economy" break**  classChoice = **"Frugal" break** |

**Error Checking Loop**

Loop variable – classSelection, classChoice

Exit Condition – classSelection == **"B"** **or** classSelection == **"E"** **or** classSelection == **"F"**

**Algorithm**

*# This function to choosing seat type***def** chooseSeat():  
 **while True**:  
 print(**"Please choose the seat type. Choosing the middle seat will deduct 25 from the total fare."**)  
 print(**"(W)indow + $75"**)  
 print(**"(A)isle + $50"**)  
 print(**"(M)iddle - $25"**)  
 seatSelection = input()  
 seatSelection = seatSelection.upper()  
 **if** seatSelection == **"W"**:  
 seatType = **"Window"  
 break  
 if** seatSelection == **"A"**:  
 seatType = **"Aisle"  
 break  
 if** seatSelection == **"M"**:  
 seatType = **"Middle"  
 break  
 else**:  
 print(**"Invalid Selection"**)  
 **return** seatType

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
|  | Get seatSelection from user | Return seatType |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| Choose seatType (W)indow, (A)isle, and (M)iddle  Window  Aisle  Middle | While = True  seatSelection == **"W"**  seatSelection == **"A"**  seatSelection == **"M"** | print(**"Please choose the seat type. Choosing the middle seat will deduct 25 from the total fare."**) print(**"(W)indow + $75"**) print(**"(A)isle + $50"**) print(**"(M)iddle - $25"**) seatSelection = input() seatSelection = seatSelection.upper()  seatType = **"Window" break**  seatType = **"Aisle" break**  seatType = **"Middle" break** |

**Error Checking Loop**

Loop variable – seatSelection, seatType

Exit Condition – seatSelection == **"W" or** seatSelection == **"A" or** seatSelection == **"M"**

**Algorithm**

*# This funciton is to input passanger age and there is no negative number***def** ageInput():  
 **while True**:  
 passangerAge = int(input(**"How old is the person travelling. Travellers under 16 years old will receive a 50% discount for the child fare."**))  
 **if** passangerAge<0:  
 print(**"Please enter a positive number"**)  
 **else**:  
 **break  
 return** passangerAge

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
|  | Get the passangerAge from the user | Return passangerAge |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| discountAge  Age | While = True  passangerAge<0 | passangerAge = int(input(**"How old is the person travelling. Travellers under 16 years old will receive a 50% discount for the child fare."**))  print(**"Please enter a positive number"**) |

**Error Checking Loop**

Loop variable – passangerAge

Exit Condition – assangerAge<0

**Algorithm**

*# This function is giving discount for poeple < 16 years old***def** discountCheck(age, price):  
 *""" The discount function """* **if** age < 16: *# Age < 16 can get discount 50% off* price = price / 2  
 print(**"Age: "** + str(age) + **" (Eligible for child ticket)"**)  
 **else**:  
 print(**"Age: "** + str(age) + **"(Not eligible for child ticket)"**)  
 **return** price

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
| Age  Price | Price = price / 2 | Return price |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| Count discount price | age < 16 | price = price / 2 print(**"Age: "** + str(age) + **" (Eligible for child ticket)"**) |

**Error Checking Loop**

Loop variable – price

Exit Condition – age

**Algorithm**

*# This function is to count the total Price of the ticket***def** countTotal(ticketSelection, flight, classChoice, seatType):  
 *""" To calculate the total from ticketSelection, flight, classChoice, and seatType """* **if** ticketSelection == **"Return"**:  
 **if** flight == **"Cairns"**:  
 price = 400  
 **elif** flight == **"Sydney"**:  
 price = 575  
 **elif** flight == **"Perth"**:  
 price = 700  
 **elif** ticketSelection == **"One-Way"**:  
 **if** flight == **"Cairns"**:  
 price = 250  
 **elif** flight == **"Sydney"**:  
 price = 420  
 **elif** flight == **"Perth"**:  
 price = 510  
 **if** classChoice == **"Business"**:  
 price += 275  
 **elif** classChoice == **"Economy"**:  
 price += 25  
 **elif** classChoice == **"Frugal"**:  
 price = price  
 **if** seatType == **"Window"**:  
 price +=75  
 **elif** seatType == **"Aisle"**:  
 price += 50  
 **elif** seatType == **"Middle"**:  
 price -= 25  
 **return** price

**IPO**

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
| ticketSelection  flight  classChoice  seatType |  | Return price |

**Apply destination decisions:**

|  |  |  |
| --- | --- | --- |
| **Path** | **Condition** | **Action** |
| ticketReturn for Cairns, Sydney, and Perth  Cairns ticketReturn  Sydney ticketReturn  Perth tickerReturn  Ticket One-Way for Cairns, Sydney, and Perth | ticketSelection == **"Return"**  flight == **"Cairns"**  flight == **"Sydney"**  flight == **"Perth"**  ticketSelection == **"One-Way"**  flight == **"Cairns"**  flight == **"Sydney"**  flight == **"Perth"**  classChoice == **"Business"**  classChoice == **"Economy"**  classChoice == **"Frugal"**  seatType == **"Window"**  seatType == **"Aisle"**  seatType == **"Middle"** | **if** flight == **"Cairns"**: **elif** flight == **"Sydney"**: **elif** flight == **"Perth**  price = 400  price = 575  price = 700  **if** flight == **"Cairns"**  **elif** flight == **"Sydney"** **elif** flight == **"Perth"**  price = 250  price = 420  price = 510  price += 275  price += 25  price = price  price +=75  price += 50  price -= 25 |

**Error Checking Loop**

Loop variable – price, ticketSelection, flight, classChoice

Exit Condition – ticketSelection == **"Return"** **or** ticketSelection == **"One-Way" or** flight == **"Cairns" or** flight == **"Sydney"** or flight == **"Perth" or** classChoice == **"Business" or** classChoice == **"Economy" or** classChoice == **"Frugal" or** seatType == **"Window" or** seatType == **"Aisle" or** seatType == **"Middle"**