Jordan Lewis
Mr J Lane
12SDD
3 June 2016

Tax Helper

Table of Contents

Problem Statement

Accountants are too expensive and slow at their work. 😬

Using an intuitively designed application is much easier than manually hand working the sums. Filling in a form which adapts as needed is easier than figuring out which section to fill in inside a form where all fields are visible.
There needs to be a tool to aid the user in this daunting task.

Method of Solution

Input	Process	Output
first name, last name, total income, phone number, bank account number, medicare number	fname = first name lname = last name … Format and print the entered information in the format: Field name: value	First Name: <fname> Last Name: <lname> Phone Number: <phone> Bank Account Number: <bank> Medicare Number: <medi>
street address 1, street address 2, suburb, post code, state	str1 = street address 1 str 2 = street address 2 … Format and print the entered information in the format: Field name: value	Street Address 1: <str1> Street Address 2: <str2> Suburb: <suburb> Post Code: <post> State: <state>
kms driven for work, cost of public transport used for work, other transport expenses for work	kms = kms driven for work * $0.66 public = cost of public transport used for work other = other transport expenses for work Format and print the entered information in the format: Field name: value	Total deductible from driving for work: $<kms> Claimable from Public Transport Costs: $<public> Claimable from Other Expenses: $<other>
number of loads of laundry with only work clothes, number of combined loads of laundry with both work and personal clothes, any costs incurred for paying for external laundry, the items that had to be bought for work	work = number pf work only laundry * $1 combined = number of combined laundry * $0.5 homeDeductible = work + combined external = costs incurred for external laundry for work clothingLetters = map items that had to be bought to letters [C, N, S, P]	Clothing letters for items bought: <clothingLetters> Total deductible from laundry: $<homeDeductible> Claimable from external laundry: $<external>
the amount of money donated to an approved organisation (if applicable)	Format and print the entered information in the format: Field name: value	Claimable from donations: $<donation>

	totalIncome = total income
	totalDeductible = kms + public + other + homeDeductible + donation
	taxableIncome = totalIncome - totalDeductible
	taxPayable = get tax payable from tax table
	medicareLevy = taxableIncome * 0.02
		Total Income: $<totalIncome>
		Total Amount Deductible: $<totalDeductible>
		Taxable Income: $<taxableIncome>
		Medicare Levy: $<medicareLevy>

Algorithm

```
BEGIN
  INPUT fName, lName, income, phNum, bankAccNum, medicareNum
  INPUT strAddr1, strAddr2, suburb, postcode, state
  INPUT numWorkLaundry, numCombLaundry, externalLaundryCost,
  typesWorkClothesBought
  INPUT totalKms, publicTransportCost, otherTransportRelatedExpenses
  INPUT amountDonated

  BEGIN calculateTaxPayable(taxableIncome)
    IF taxableIncome <= 18200:
      RETURN 0
    ENDIF
    IF 18201 <= taxableIncome <= 37000:
      RETURN (taxableIncome – 18200) * 0.19
    ENDIF
    IF 37001 <= taxableIncome <= 80000:
      RETURN ((taxableIncome – 37000) * 0.325) + 3572
    ENDIF
    IF 80001 <= taxableIncome <= 180000:
      RETURN ((taxableIncome – 80000) * 0.37) + 17547
    ENDIF
    IF taxableIncome >= 180001:
      RETURN ((taxableIncome – 180000) * 0.47) + 54547
    ENDIF
  END calculateTaxPayable

  BEGIN printLine()
    line = “”
    FOR i in 0 TO 25
      line = line + “–“
    NEXT i
    PRINT line
  END printLine

  BEGIN mapClothingItemsToLetters(items)
    ‘Listboxes in Tkinter return the indexes of what was selected
    boughtCodes = “”
    codes = {
      “Compulsory Work Uniform” : “C”,
      “Non-compulsory Work Uniform” : “N”,
      “Occupation Specific Clothing” : “S”,
      “Protective Clothing” : “P”,
    }
    FOR item IN items
      key = codes.keys()(item)
      boughtCodes = boughtCode + codes[key] + “, “
    NEXT item
    boughtCodes = boughtCodes and remove the last two characters
    RETURN boughtCodes
  END mapClothingItemsToLetters

  totalDrivingCost = totalKms * 0.66
  totalLaundryCost = numWorkLaundry + (numCombLaundry * 0.5)
  totalDeductible = totalDrivingCost + totalLaundryCost +
  publicTransportCost + otherTransportRelatedExpenses + amountDonated

  totalIncome = income
  taxableIncome = totalIncome – totalDeductible

  PRINT “First Name: “ + fName
  PRINT “Last Name: “ + lName
  PRINT “Phone Number: “ + phNum
  PRINT “Bank Account Number: “ + bankAccNum
  PRINT “Medicare Number: “ + medicareNum
  printLine()
  PRINT “Street Address 1: “ + strAddr1
  PRINT “Street Address 2: “ + strAddr2
  PRINT “Suburb: “ + suburb
  PRINT “Post Code: “ + postcode
  PRINT “State: “ + state
  printLine()
  PRINT “Total deductible from driving for work: “ + totalDrivingCost
  PRINT “Claimable from Public Transport Costs: “ + publicTransportCost
  PRINT “Claimable from Other Expenses: “ + otherTransportRelatedExpenses
  printLine()
  PRINT “Clothing letters for items bought: “ +
  mapClothingItemsToLetters(typesWorkClothesBought)
  PRINT “Total deductible from laundry: $” + totalLaundryCost
  PRINT “Claimable from external laundry: $” + externalLaundryCost
  printLine()
  PRINT “Claimable from donations: $” + amountDonated
  printLine()
  PRINT “Total Income: $” + totalIncome
  PRINT “Total deductible: $” + totalDeductible
  PRINT “Taxable Income: $” + taxableIncome
  PRINT “Tax Payable: $” + calculateTaxPayable(taxableIncome)
  PRINT “Medicare Levy: $” + taxableIncome * 0.02
END
```

Test Data

Due to the purely mathematical nature of this program, there only needs to be one test to see if it is indeed correct. Otherwise, there should be error catching such as input validation.

Inputs	Expected Result
Taxable Income of “100”	0
Taxable Income of “18200”	0
Taxable Income of “18201”	0.19
Taxable Income of “19000”	152
Taxable Income of “37000”	3572
Taxable Income of “37001”	3572.325
Taxable Income of “38000”	3897
Taxable Income of “80000”	17547
Taxable Income of “80001”	17547.37
Taxable Income of “81000”	17917
Taxable Income of “180000”	54547
Taxable Income of “180001”	54547.47
Taxable Income of “200000”	63947
<blank> into a visible field	Incomplete Form Error
“a” into a numerical field	Invalid Data Type
“1a” into a numerical field	Invalid Data Type
" “ into a required field	Incomplete Form Error
<Any string> into a string input i.e. “Jordan”	No Error
<Any number> into a numerical field i.e. “1” or “2.2”	No Error, rounding if the required data type is an integer
“200” in the kms driven	Claimable from driving for work: $132
“50” in work only loads and “300” in combined loads	Claimable from laundry: $200
Select “Compulsory Work Uniform” in the listbox	C
Select "Compulsory Work Uniform” and “Protective Clothing” in the listbox	C, P