Alexander Laudino CSC 161-04

Feb. 21, 2021 Dr. Farrett

Lab Assignment 4

Internet Service Provider

// Import JOptionPane for message dialog boxes

// Main method

// While loop to validate user package letter input

// packageLetter = User input

// If packageLetter = A, B, or C

// Switch statements to execute code based on package letter case

// Case for package letter A, calls packageA method, display output

// Case for package letter B, calls packageB method, display output

// Case for package letter C, uses packageC variable, display output

// Break out of letter validation loop

// Else packageLetter != A, B, or C, continues loop until valid input entered

// Package A method, returns total bill based on user hours input

// While loop to validate user input is a double type

// Try hours = User input, convert input to double type

// Catch hour != double type

// If hours <= 10, return 9.95

// Else, return 9.95 + (2 \* (hours - 10))

// Package B method, returns total bill based on user hours input

// While loop to validate user input is a double type

// Try hours = User input, convert input to double type

// Catch hours != double type

// If hours <= 20, return 13.95

// Else, return 13.95 + (1 \* (hours – 20))

Internet Service Provider 2

// Import JOptionPane for message dialog boxes

// Main method

// Declare and initialize variables

// double package = 19.95

// double hours = 0

// While loop to validate user package letter input

// packageLetter = User input

// If packageLetter = C, display output

// Else packageLetter = B or packageLetter = C

// While loop to validate user input is a double type

// Try hours = input, convert input to double type

// Switch statement for Package A or B

// Case for Package A, calls Package A method and passes hours variable to calculate bill

and display output

// If Package A bill > Package B price and Package A bill <= Package C price,

Package A bill – package B price = Savings for B, display output

// Else if Package A bill > Package B price and Package A bill > Package C price,

Package A bill – Package B price = Savings for B, display output

Package A bill – Package C price = Savings for C, display output

// Case for Package B, calls Package B method and passes hours variable to calculate bill

and display output

// If Package B bill > Package C price,

Package B bill – Package C price = Savings for C, display output

// Break out of double validation loop

// Catch hours != double type

// Break out of package letter validation loop

// Package A method, takes hours as parameter, return total bill

// If hours <= 10, return 9.95

// Else, return 9.95 + (2 \* (hours - 10))

// Package B method, takes hours as parameter, return total bill

// If hours <= 20, return 13.95

// Else, return 13.95 + (1 \* (hours – 20))