

**Developer**: Tiffany McDonnell

**Date**: 7-23-2022

# IT 145 Global Rain Summary Report Template

## **Pseudocode**

**START** (Check out)

**INPUT** pet type

**INPUT** days pet had boarded

**IF** pet type is cat

**CALCULATE** amountDue equals 18.00 \* daysStay

**PRINT** amountDue

catSpace **INCRIMENTS** by 1

**IF** pet type is dog

**IF** dogWeight < 20

**IF** grooming **AND** daysStay equals 2 or more

**CALCULATE** amountDue equals 19.95 + the product of (24 \* daysStay)

**ELSE**

**CALCULATE** amuntDue equals 24.00 \* daysStay

**ELSE** **IF** dogWeight < 30

**IF** grooming **AND** daysStay equals 2 or more

**CALCULATE** amountDue equals 24.96 + the product of (29 \* daysStay)

**ELSE**

**CALCULATE** amuntDue equals 29.00 \* daysStay

**ELSE**

**IF** grooming **AND** daysStay equals 2 or more

**CALCULATE** amountDue equals 29.95 + the product of (34 \* daysStay)

**ELSE**

**CALCULATE** amuntDue equals 34.00 \* daysStay

**PRINT** amoutDue

dogSpace **INCRIMENTS** by 1

**END**

## **Flowchart**

dogSpace

Increments by 1

Yes and stayed at least 2 days

Yes and stayed at least 2 days

Yes and stayed at least 2 days

Yes but stayed less than 2 days

Yes but stayed less than 2 days

Yes but stayed less than 2 days

No

No

No

30lb and greater

20lb-29lb

Less than 20lb

Cat

Dog

amountDue =

(29.00 \* DaysStay)

amountDue =

(34.00 \* DaysStay)

amountDue =

(24.00 \* DaysStay)

catSpace

Increments by 1

amountDue =

18 \* DaysStay

Get

Grooming

Get

Grooming

Get

Grooming

Get

Dog Weight

Get

Pet Type

Get

Days Staying

Get

Pet Name

amountDue =

29.95 +

(34 \* DaysStay)

amountDue =

24.95 +

(29 \* DaysStay)

amountDue =

19.95 +

(24 \* DaysStay)

End

Print

amountDue

Start

## **OOP Principles Explanation**

Within this project I have utilized encapsulation by keeping all my attributes private within the classes. The classes can access the public methods to read in the information needed. One way abstraction in achieved is how Pet Class can’t see that dog’s weight that is being used to calculate the amount due that Pet Class receives. I didn’t really use inheritance in this project but, I know I could of to allow the classes to interact with each other to achieve a functional program. Polymorphism is used in this case to allow no matter what sub class is being called to still utilize the parent classes methods.