

**Developer**: Jen Nadeau

**Date**: 3/25/2022

# IT 145 Global Rain Summary Report Template

**Pseudocode for Pet Check-In:**

INPUT pet type, either dog or cat

SET pet type to either dog or cat from input

IF pet type is dog THEN

IF dog spaces less than 30 THEN

INPUT pet name

SET pet name

IF new pet THEN

INPUT pet age

SET pet age

INPUT pet weight

SET pet weight

ELSE

OUTPUT pet age and pet weight

INPUT is information correct

IF incorrect THEN

INPUT pet age

SET pet age

INPUT pet weight

SET pet weight

ELSE

CONTINUE with current pet age and weight

ENDIF

ENDIF

INPUT length of stay

SET length of stay

IF length of stay greater than or equal to 2 THEN

INPUT if owner would like pet to be groomed

SET grooming to either yes or no from input

ELSE

OUTPUT grooming not offered

ENDIF

ASSIGN pet to space

ELSE

OUTPUT dog spaces full, no room to board

ENDIF

ELSE

IF cat spaces less than 12 THEN

INPUT pet name

SET pet name

IF new pet THEN

INPUT pet age

SET pet age

ELSE

CONTINUE with current pet age

ENDIF

INPUT length of stay

SET length of stay

ASSIGN cat to space

ELSE

OUTPUT cat spaces full, no room to board

ENDIF

ENDIF

**Flowchart for Pet Check-In:**

Diagram

Description automatically generated

END

END

END

END

END

## OOP Principles Explanation

Briefly explain how you applied object-oriented programming principles and concepts (such as encapsulation, inheritance, and so on) in your software development work thus far. Your explanation should be one paragraph, or four to six sentences.

The four object-oriented principles (encapsulation, abstraction, inheritance, and polymorphism) have been applied to the software development work for Pet BAG so far. By having three different objects or classes (Pet, Dog, and Cat) and through encapsulation and abstraction, certain attributes of a particular class can be kept private so other classes can call them but cannot directly modify them. The child classes (Dog and Cat) are derived from the parent class (Pet). By applying inheritance and polymorphism to extend the parent class to the child classes, the child classes can be separated with logic that is only necessary for them but can use the common logic of the parent class. By applying OOP, we can help ensure the understanding and ease of maintenance of code for all other developers working on the app.