Class Pet

Properties:

petType

petName

petAge

daysStay

grooming

spaceId

amountDue

Methods:

Constructor(petType, petName, petAge, daysStay)

Initialize petType, petName, petAge, daysStay

Set grooming to false

Set spaceId to null

Set amountDue to 0.0

SetGrooming(groomingOption)

if petType is "Dog" and daysStay >= 2 then

if groomingOption is true then

Set grooming to true

Add grooming fee to amountDue based on petType

else

Set grooming to false

Subtract grooming fee from amountDue based on petType

end if

end if

SetSpaceId(spaceId)

Set spaceId to spaceId

GetAmountDue()

Return amountDue

End Class

Method CheckInPet

Input: petType, petName, petAge, daysStay

Output: successMessage or errorMessage

Initialize dogSpaces = 30

Initialize catSpaces = 12

if petType is "Dog" and dogSpaces > 0 then

Create a new Pet object with petType, petName, petAge, daysStay

Assign a spaceId for the Dog

Decrement dogSpaces

if daysStay >= 2 then

Ask for grooming option

if yes, call SetGrooming(true) on the pet object

end if

return successMessage

else if petType is "Cat" and catSpaces > 0 then

Create a new Pet object with petType, petName, petAge, daysStay

Assign a spaceId for the Cat

Decrement catSpaces

return successMessage

else

return errorMessage

end if

End Method

Method CheckOutPet

Input: pet (a Pet object)

Output: totalFees

if pet is not null then

Calculate totalFees based on pet's petType, daysStay, and grooming status

Increment dogSpaces if pet is a Dog, or catSpaces if pet is a Cat

Set pet's spaceId to null

Set pet's amountDue to 0.0

return totalFees

else

return errorMessage

end if

End Method