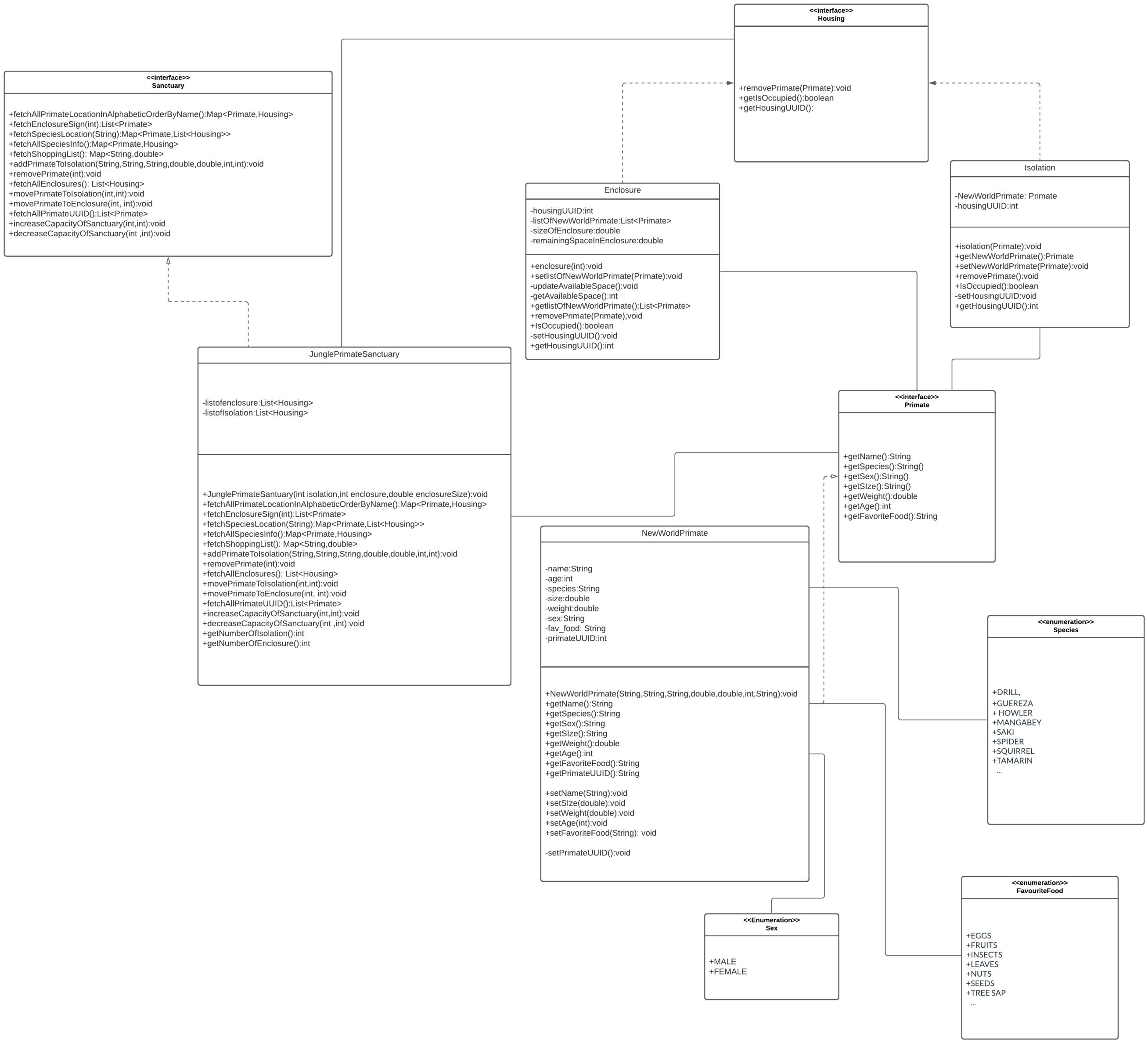

```
<<interface>>
Sanctuary

+fetchAllPrimateLocationInAlphabeticOrderByName():Map<Primate,Housing>
+fetchEnclosureSign(int):List<Primate>
+fetchSpeciesLocation(String):Map<Primate,List<Housing>>
+fetchAllSpeciesInfo():Map<Primate,Housing>
+fetchShoppingList(): Map<String,double>
+addPrimateToIsolation(String,String,String,double,double,int,int):void
+removePrimate(int):void
+fetchAllEnclosures(): List<Housing>
+movePrimateToIsolation(int,int):void
+movePrimateToEnclosure(int, int):void
+fetchAllPrimateUUID():List<Primate>
+increaseCapacityOfSanctuary(int,int):void
+decreaseCapacityOfSanctuary(int ,int):void
```

```
<<interface>>
Housing

+removePrimate(Primate):void
+getIsOccupied():boolean
+getHousingUUID():
```

```
Enclosure

-housingUUID:int
-listOfNewWorldPrimate:List<Primate>
-sizeOfEnclosure:double
-remainingSpaceInEnclosure:double

+enclosure(int):void
+setlistOfNewWorldPrimate(Primate):void
-updateAvailableSpace():void
-getAvailableSpace():int
+getlistOfNewWorldPrimate():List<Primate>
+removePrimate(Primate);void
+IsOccupied():boolean
-setHousingUUID():void
+getHousingUUID():int
```

```
Isolation

-NewWorldPrimate: Primate
-housingUUID:int

+isolation(Primate):void
+getNewWorldPrimate():Primate
+setNewWorldPrimate(Primate):void
+removePrimate():void
+IsOccupied():boolean
-setHousingUUID:void
+getHousingUUID():int
```

```
JunglePrimateSanctuary

-listofenclosure:List<Housing>
-listofIsolation:List<Housing>

+JunglePrimateSantuary(int isolation,int enclosure,double enclosureSize):void
+fetchAllPrimateLocationInAlphabeticOrderByName():Map<Primate,Housing>
+fetchEnclosureSign(int):List<Primate>
+fetchSpeciesLocation(String):Map<Primate,List<Housing>>
+fetchAllSpeciesInfo():Map<Primate,Housing>
+fetchShoppingList(): Map<String,double>
+addPrimateToIsolation(String,String,String,double,double,int,int):void
+removePrimate(int):void
+fetchAllEnclosures(): List<Housing>
+movePrimateToIsolation(int,int):void
+movePrimateToEnclosure(int, int):void
+fetchAllPrimateUUID():List<Primate>
+increaseCapacityOfSanctuary(int,int):void
+decreaseCapacityOfSanctuary(int ,int):void
+getNumberOfIsolation():int
+getNumberOfEnclosure():int
```

```
<<interface>>
Primate

+getName():String
+getSpecies():String()
+getSex():String()
+getSIze():String()
+getWeight():double
+getAge():int
+getFavoriteFood():String
```

```
NewWorldPrimate

-name:String
-age:int
-species:String
-size:double
-weight:double
-sex:String
-fav_food: String
-primateUUID:int

+NewWorldPrimate(String,String,String,double,double,int,String):void
+getName():String
+getSpecies():String
+getSex():String
+getSIze():String
+getWeight():double
+getAge():int
+getFavoriteFood():String
+getPrimateUUID():String

+setName(String):void
+setSIze(double):void
+setWeight(double):void
+setAge(int):void
+setFavoriteFood(String): void

-setPrimateUUID():void
```

```
<<enumeration>>
Species

+DRILL,
+GUEREZA
+ HOWLER
+MANGABEY
+SAKI
+SPIDER
+SQUIRREL
+TAMARIN
  ...
```

```
<<Enumeration>>
Sex

+MALE
+FEMALE
```

```
<<enumeration>>
FavouriteFood

+EGGS
+FRUITS
+INSECTS
+LEAVES
+NUTS
+SEEDS
+TREE SAP
  ...
```

Test Cases	Input/strategy	Expected Value
TestPrimateForNegativeweight	NewWorldPrimate(John,drill,male,40,-43.5,10,eggs)	IllegalArgumentException
TestPrimateForNegativeAge	NewWorldPrimate(John,drill,male,40,53.5,-11,eggs)	IllegalArgumentException
TestPrimateForZeroWeight	NewWorldPrimate(John,drill,male,40,0,10,eggs)	IllegalArgumentException
TestPrimateForInvalidSex	NewWorldPrimate(John,drill,krill,40,-43.5,10,eggs)	EnumConstantNotPresentException
TestPrimateForInvalidSpecies	NewWorldPrimate(John,apes,male,40,-43.5,10,eggs)	EnumConstantNotPresentException
TestPrimateForValidParameter	NewWorldPrimate(John,drill,male,40,68,10,eggs)	Successful creation of object
TestPrimateForInvalidFavouriteFood	NewWorldPrimate(John,drill,male,40,-43.5,10,spinach)	EnumConstantNotPresentException
TestgetName()	NewWorldPrimate(John,drill,male,40,68,10,eggs)	John
TestgetSex()	NewWorldPrimate(John,drill,male,40,68,10,eggs)	male
TestgetAge()	NewWorldPrimate(John,drill,male,40,68,10,eggs)	10
TestgetSize()	NewWorldPrimate(John,drill,male,40,68,10,eggs)	40
TestgetSpecies()	NewWorldPrimate(John,drill,male,40,68,10,eggs)	drill
TestgetFavoriteFood()	NewWorldPrimate(John,drill,male,40,68,10,eggs)	eggs
Test Species Enum	assertEquals("eggs", Species.EGGS.name());	TRUE
Test Species Enum	assertEquals("fish", Species.FISH.name());	EnumConstantNotPresentException
Test Sex Enum	assertEquals("male", Sex.Male.name());	TRUE
Test Sex Enum	assertEquals("other", Sex.OTHER.name());	FALSE
TestEnclosureForNegativeSize	enclosure(-100)	IllegalArgumentException
TestSetlistOfNewWorldPrimate	Add 5 primates to an enclosure. Fetch them using getListofNewWorldPrimate. Compare the list with the input list	TRUE
TestAvailableSpace	Add n primates to an enclosure. Fetch available space using getAvailableSpace and compare with the manually calculated value	TRUE
testRemovePrimate	Add one Primate to isolation. Call removePrimate. Follow this with call to getNewWorldPrimate and check for empty list or null value	TRUE
testIsOccupied	Add 1 primate to isolation. Call isOccupied and assert for true. Move 1 primate to enclosure and call isOccupies and assert true	TRUE
TestFetchAllPrimateLocationInAlphabeticOrderByName	Add Primates to isolation and then move primates to different enclosures. Get the map by calling TestFetchAllPrimateLocationin AlphabeticalOrder. Check if the Keys of the maps are in alphabetical order and compare the map with the map which we create in the test cases as we are aware of the primates created as well the location to which they were sent.	TRUE
TestFetchShoppingList	Add Primates to isolation and move few of them to enclosures. Fetch the shopping list using FetchShoppingList and compare it with the manually calculated Map entries	TRUE
TestFetchEnclosure()	Add Primates to the isolation and move the primates to an enclosure. Fetch the list of Primate using fetchEnclosureSign and compare with the map prepared within the test cases.	TRUE
TestFetchAllSpeciesInfo	Add Primates to the enclosure and isolation. Fetch the Map of Primate using fetchAllSpeciesInfo and compare with the map prepared within the test cases manually.	TRUE
TestRemovePrimate	Add n primates to both eisolation.Choose one of the primates to delete. Call fetchAllPrimateLocationInAlphabeticOrder and check if the choosen primate is present in the returned map. Then call removePrimate. Call fetchAllPrimateLocationInAlphabeticOrder and check for absense of the chosen Primate	TRUE
TestSpeciesLocation	Add primates of different species to isolation. Call speciesLocation and compare the map with the map in the Test case created manually in the test function.	TRUE
TestAddPrimateToIsolation	Add n primates to isolation. Fetch the primates and their location using fetchAllPrimateLocationInAlphabeticOrderByName().Compare the primate UUID and its location with the map that we have prepared manually	TRUE
TestAddPrimateToIsolation	Add n primates where n is more than the initial capacity of the sanctuary.	IndexOutOfBoundsException
TestMovePrimateToEnclosure/TestMovePrimateToisolation	Add n primates to the isolation. Call movePrimateToEnclosure(int,int). Call fetchAllPrimateLocationInAlphabeticOrderByName() to get the Map of primate and location. Compare the primate and their housing with the entries of the map created manually for this purpose.	TRUE
TestMovePrimateToisolation	Move a primate that does not exist	IllegalArgumentException
TestMovePrimateToEnclosure	Move a primate that does not exist	IllegalArgumentException
TestMovePrimateToisolation	Move a primate to an isolation which already has a primate	IllegalStateException
TestMovePrimateToEnclosure	Move a primate to an enclosure which is already full	IllegalStateException
TestincreaseCapacityOfSanctuary	call increaseCapacityOfSanctuary(int,int). Call getNumberOfIsolation() and getNumberOfEnclosure() and compare the individual value with the value that is set in increaseCapacityOfSanctuary	TRUE
TestdecreaseCapacityOfSanctuary	call decreaseCapacityOfSanctuary(int,int). Call getNumberOfIsolation() and getNumberOfEnclosure() and compare the individual value with the value that is set in decreaseCapacityOfSanctuary	TRUE