

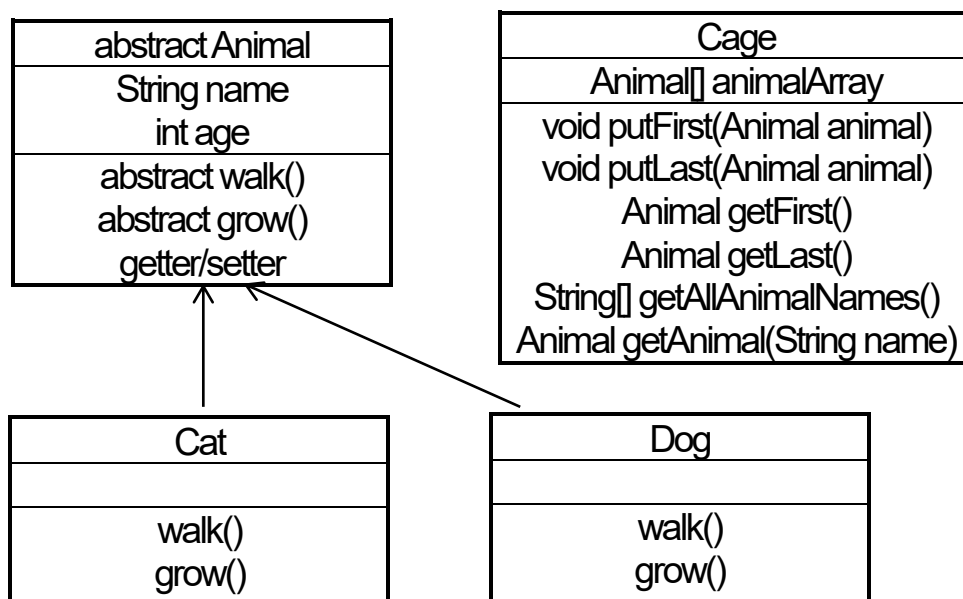
## Java 基礎問題-02-02-09

以下のクラス図を参考にして動物をあらわす抽象クラス **Animal** と檻をあらわす **Cage** クラスをモデリングし、Java で実装してください。

クラス図に不足している情報があれば考えて補ってください。

KadaiCage9.java に main() メソッドを作成し動作を確認してください。

Animal クラスは直接インスタンスを生成しないこととし、Animal クラスを継承した Dog や Cat などのサブクラスを作って操作するようにします。



## ヒント

### ▼必要とされる知識

継承

抽象クラス

オーバーライド

### ▼考え方のポイント

Java クラスモデリング問題 5 を使用し、効率的に実装を進めましょう。抽象クラスの実装内容などは no009 パッケージに格納されている Java ファイルの Javadoc コメント(`/** */`)に記載されていますので、そちらを参照し実装していきましょう。

この問題は、練習問題 05 で作成した Cat クラスが Animal クラスのサブクラスとしてモデリングされたものと考えれば良いでしょう。

Cage クラスの `getFirst()`, `getLast()` メソッドの戻り値は Animal 型になっています。つまり檻の中の動物が Dog であっても Cat であっても、これらのメソッドにより Animal として取得することができるようになります。

## 実行結果 ※数値と記号は半角を利用してください。

※以下実行結果の通りコンソールに出力されるように実装しましょう。

===== 籠に動物を入れます (putFirst (Animal animal)、putLast (Animal animal) メソッド) =====

たま1をかごの先頭に入れました

ポチ1をかごの末尾に入れました

たま2をかごの先頭に入れました

ポチ2をかごの末尾に入れました

たま3をかごの先頭に入れました

ポチ3をかごの末尾に入れました

たま4をかごの先頭に入れました

ポチ4をかごの末尾に入れました

たま5をかごの先頭に入れました

ポチ5をかごの末尾に入れました

かごがいっぱいなので動物は入りません。

===== 籠に動物を入れました (putFirst (Animal animal)、putLast (Animal animal) メソッド) =====

===== 今の籠の中身を一覧で表示開始 (getAllAnimalNames () メソッド) =====

たま5 たま4 たま3 たま2 たま1 ポチ1 ポチ2 ポチ3 ポチ4 ポチ5

===== 今の籠の中身を一覧で表示終了 (getAllAnimalNames () メソッド) =====

===== 先頭の動物と、末尾の動物を取得開始 (getFirst () メソッド、getLast () メソッド) =====

先頭の動物はたま5です

末尾の動物はポチ5です

===== 先頭の動物と、末尾の動物を取得終了 (getFirst () メソッド、getLast () メソッド) =====

===== 特定の動物をみつけます (getAnimal (String name) メソッド、grow ()、walk () メソッド) =====

猫のたま1が吠えた

現在たま1は1才です

たま1は2才になりました

犬のポチ4が吠えた

現在ポチ4は4才です

ポチ4は5才になりました

===== 特定の動物をみつめました (getAnimal (String name) メソッド、grow ()、walk () メソッド) =====

