

Quais linhas em Girino.java apresentam erro de compilador? (Escolha todas as opções aplicáveis.)

```
// Sapo.java
1: package animal;
2: public class Sapo {
3:     protected void coaxar() { }
4:     void saltar() { }
5: }
// Girino.java
1: package other;
2: import animal.*;
3: public class Girino extends Sapo {
4:     public static void main(String[] args) {
5:         Girino t = new Girino();
6:         t.coaxar();
7:         t.saltar();
8:         Sapo f = new Girino();
9:         f.coaxar();
10:        f.saltar();
11:    } }
```

- A. Linha 5
- B. Linha 6
- C. Linha 7
- D. Linha 8
- E. Linha 9
- F. Linha 10
- G. Todas as linhas são compiladas.

ANÁLISE DA QUESTÃO

Vamos reavaliar cada linha no arquivo Girino.java:

6: t.coaxar();

- método `coaxar` é *protected*. Como *Girino* é uma subclasse de *Sapo*, essa linha compila sem problemas. Não há erro aqui.

7: t.saltar();

- O método `saltar()` tem visibilidade de pacote (*default*). Isso significa que ele **só é acessível dentro do mesmo pacote** (*animal*). Como *Girino* está em um pacote diferente (*other*), essa linha gera um **erro de compilação**.

9: f.coaxar();

- A variável *f* é do tipo *Sapo*, mas instanciado como *Girino*. O método `coaxar()` é *protected*, então ele só pode ser acessado a partir de uma subclasse ou no mesmo pacote. No entanto, como *f* é uma referência de *Sapo* e não uma referência de sua subclasse *Girino*, essa linha gera um **erro de compilação**.

10: f.saltar();

- O método `saltar()` tem visibilidade de pacote (*default*). Assim como na linha 7, isso gera um erro de compilação porque o método `saltar()` não é acessível fora do pacote *animal*, essa linha gera um **erro de compilação**.

As linhas que geram erro de compilação são:

- C. Linha 7
- E. Linha 9
- F. Linha 10

Vamos falar um pouco mais sobre os modificadores de acesso em Java. Podemos montar a seguinte tabela, descrevendo os modificadores de acesso:

Modificadores de Acesso em Java

Modificador	Classe	Pacote	Subclasse (mesmo pacote)	Subclasse (pacote diferente)	Fora do Pacote
public	Sim	Sim	Sim	Sim	Sim
protected	Sim	Sim	Sim	Sim	Não
default (sem modificador)	Sim	Sim	Sim	Não	Não
private	Sim	Não	Não	Não	Não

A especificação em §6.6 do *Java Language Specification (JLS)* detalha alguns pontos:

- ✓ **public:** "If the member or constructor is declared public, then access is permitted".
[“Se o membro ou construtor for declarado *public*, o acesso será permitido”]
- ✓ **protected:** "If the member or constructor is declared protected, then access is permitted only when one of the following is true: Access to the member or constructor occurs from within the package containing the class in which the protected member or constructor is declared. Access is correct as described in §6.6.2".
[“Se o membro ou construtor for declarado *protected*, o acesso será permitido somente quando uma das seguintes opções for verdadeira: O acesso ao membro ou construtor ocorre de dentro do pacote que contém a classe na qual o membro ou construtor protegido é declarado. O acesso está correto conforme descrito em §6.6.2”]
A seção §6.6.2 esclarece que o modificador *protected* é projetado para permitir que subclasses acessem membros e construtores *protected* dos objetos que estão implementando, **mesmo se estiverem em pacotes diferentes**. Isso ajuda a manter um nível de encapsulamento enquanto ainda permite que as subclasses estendam a funcionalidade das classes parentais.
- ✓ **default:** "If the member or constructor is declared with package access, then access is permitted only when the access occurs from within the package in which the class, interface, type parameter, or reference type is declared. A class member or constructor declared without an access modifier implicitly has package access".
[Se o membro ou construtor for declarado com acesso ao pacote, o acesso será permitido somente quando o acesso ocorrer de dentro do pacote no qual a classe, interface, parâmetro de tipo ou tipo de referência for declarado. Um membro de classe ou construtor declarado sem um modificador de acesso tem implicitamente acesso ao pacote]
- ✓ **private:** "If the member or constructor is declared private. Access is permitted only when one of the following is true: Access occurs from within the body of the top level class or interface that encloses the declaration of the member or constructor".
[Se o membro ou construtor for declarado *private*. O acesso é permitido somente quando uma das seguintes condições for verdadeira: O acesso ocorre dentro do corpo da classe ou interface de nível superior que inclui a declaração do membro ou construtor]