

Escolhendo o alvo

Como agora precisamos dizer em qual alvo estamos atirando, vamos modificar o método `Atira` para receber o alvo como parâmetro:

```
public class Torre : MonoBehaviour
{
    private void Atira (Inimigo inimigo)
    {
        float tempoAtual = Time.time;
        if (tempoAtual > momentoDoUltimoDisparo + tempoDeRecarga)
        {
            momentoDoUltimoDisparo = tempoAtual;
            GameObject pontoDeDisparo =
                this.transform.Find ("CanhaoDaTorre/PontoDeDisparo").gameObject;
            Vector3 posicaoDoPontoDeDisparo =
                pontoDeDisparo.transform.position;
            GameObject projetilObject =
                (GameObject) Instantiate (projetilPrefab,
                                          posicaoDoPontoDeDisparo,
                                          Quaternion.identity);
            Missil missil = projetilObject.GetComponent<Missil>();
            missil.DefineAlvo(inimigo);
        }
    }
}
```

- Neste momento cada míssil recebe a informação de qual é o seu alvo, temos que alterar também o script `Missil` para lidar com essa nova informação:

```
public class Missil : MonoBehaviour
{
    public void DefineAlvo(Inimigo inimigo)
    {
        alvo = inimigo;
    }
}
```

- O míssil só pode ter inimigos como alvo, vamos modificar o tipo do atributo `alvo` neste mesmo script:

```
public class Missil : MonoBehaviour
{
    private Inimigo alvo;
}
```

- Além disso, como o míssil recebe o alvo diretamente da torre, não precisamos mais buscar o alvo no método `Start` :

```
public class Missil : MonoBehaviour
{
    void Start ()
    {
        // Remover a linha abaixo
        // alvo = GameObject.Find("Inimigo");

        AutoDestroiDepoisDeSegundos (10);
    }
}
```

Ao testar o jogo, qual o comportamento das torres?

Responda

INSERIR CÓDIGO		FORMATAÇÃO