

# Diagrams-The Lord of rings

## Domain Model

@startuml

class TheLordOfRings

class Team

class Battlefield

abstract Player

class Heroes

class Beasts

class Hobbits

class Elves

class Humans

class Goblins

class Orcs

TheLordOfRings--> Player

TheLordOfRings--> Battlefield

TheLordOfRings--> Team

Team --> Battlefield

Battlefield --> Player

Heroes-up-|> Player

Beasts-up-|> Player

Hobbits-up-|> Heroes

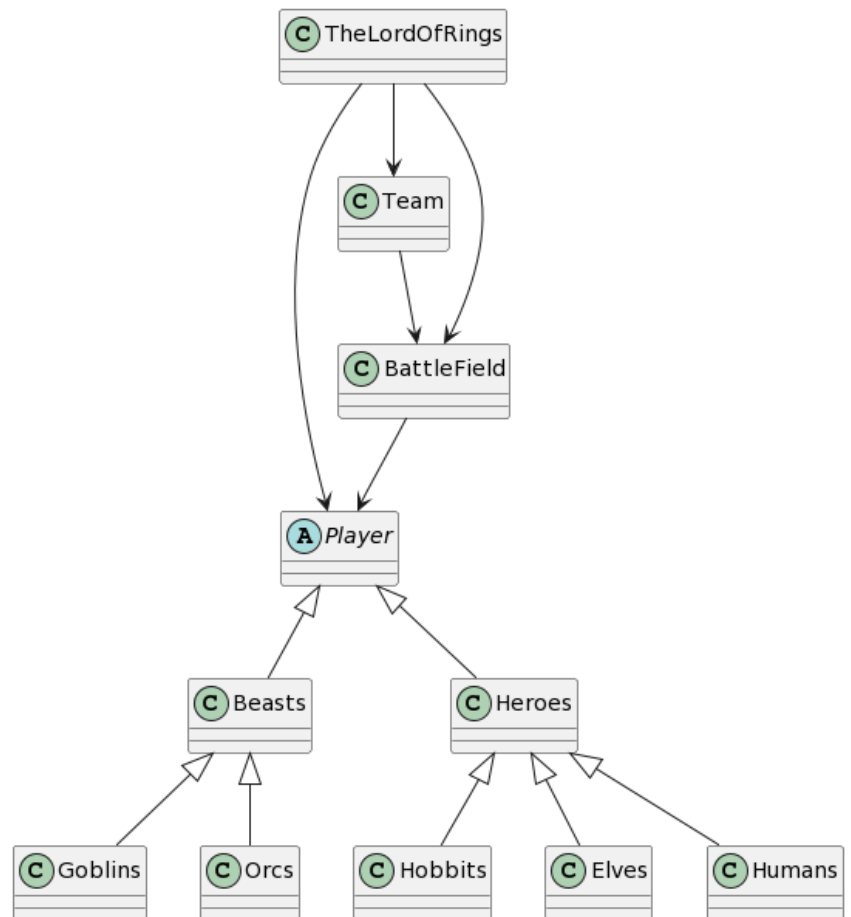
Elves-up-|> Heroes

Humans-up-|> Heroes

Goblins-up-|> Beasts

Orcs-up-|> Beasts

@enduml



## *Class Diagram*

```
@startuml
class TheLordOfRings
{
main()
}

class Team
{
-ArrayList<Player> ActivePlayers
-Team()
-FindPlayer(Player player)
-getPlayer(position)
-CheckTeam()
-PlayerLoses()
}

class Battlefield
{
-Fight()
-ContinueUser()
}
package players{

abstract Player
{
-Name
-LifePoints
-ReinforcementArmor
-Characters (Name, ReinforcementArmor)
-getName()
-setName()
-getLifePoints()
-setLifePoints()
-getReinforcementArmor()
-setReinforcementArmor()
-isDead()
-Attacked()
-Attack()
-{abstract} ToString()
}

class Heroes
{
-dice1
-dice2
-Heroes(Name, ReinforcementArmor)
-Attack()
-{abstract} ToString()
}

class Beasts
{
-dice
-Beasts(Name, ReinforcementArmor)
```

```

-Attack()
-{abstract} ToString()
}

class      Hobbits
{
-Hobbits (Name, ReinforcementArmor)
}
class      Elves
{
-Elfos (Name, ReinforcementArmor)
-Attack()
}
class      Humans
{
-Humanos (Name, ReinforcementArmor)
}
class      Goblins
{
-Trasgos(Name, ReinforcementArmor)
}
class      Orcs
{
-Orcos(Name, ReinforcementArmor)
}

```

```

}

```

```

TheLordOfRings--> Player
TheLordOfRings--> BattleField
TheLordOfRings--> Team
Team --> BattleField

```

```

BattleField --> Player

```

```

Heroes-up-|> Player
Beasts-up-|> Player

```

```

Hobbits-up-|> Heroes
Elves-up-|> Heroes
Humans-up-|> Heroes

```

```

Goblins-up-|> Beasts
Orcs-up-|> Beasts

```

```

@enduml

```

