

# 21 Výbušný šíp

V této lekci si ukážeme, jak pomocí příkazu `execute` vytvořit výbušný šíp.

## Získání informací o entitě

Pomocí příkazu `data` si můžeme zjistit, jaké informace o šípu máme dostupné

```
1 /data get entity <cíl>
```

- entita musí být **jen jedna**, proto nemůžeme použít jen samostatné `@e`, ale entitu musíme upřesnit typem, počtem, ...
  - např. pro získání informací o nejbližším šípu můžeme použít `@e[type=minercraft:arrow, limit=1]`
    - výsledek vždy musíme omezit na **1 entitu**, proto **vždy** používáme `limit=1`

Po spuštění příkazu se nám vypíše spoustu informací. Jsou to veškeré informace, které tato entita (šíp) obsahuje. Všechny tyto informace můžeme dále využít pro určení konkrétní entity. Tyto informace se označují jako `nbt data`.

```
Entita šíp má následující data entity: {Motion: [-1.0786119552513598d, -1.5548000184863806d, 1.4938508489502753d], damage: 2.0d, Owner: [I; 921201172, 1047545226, -1941626096, -636963409], PierceLevel: 0b, shake: 0b, inGround: 1b, ShotFromCrossbow: 0b, pickup: 2b, Invulnerable: 0b, LeftOwner: 1b, Air: 300s, OnGround: 0b, PortalCooldown: 0, Rotation: [-35.830605f, -39.23832f], life: 180s, inBlockState: {Name: "minecraft:netherite_block"}, FallDistance: 0.0f, SoundEvent: "minecraft:entity.arrowhit", crit: 0b, Pos: [-343.12163714728314d, 53.03162736008831d, 164.76709813196138d], HasBeenShot: 1b, Fire: -1s, UUID: [I; -441080187, 1023164520, -1770836760, -1970778911]}
```

Pro náš příklad nás bude zajímat, jestli je šíp v něčem zabodnutý. To nám říká vlastnost `inGround`.

- je typu *pravdivostní hodnota* ( `boolean` )
- má hodnotu **0** - **nepravda** nebo **1** - **pravda**

## Výběr entity pomocí `nbt data`

Tyto informace můžeme dále použít k výběru entity a to tak, že je uvedeme jako hodnotu pro značku `nbt` - např:

```
1 @e[nbt={vlastnost:hodnota}]
```

- dvojici uvádíme stejně, jak je uvedená ve výpisu vlastností - tedy i s **typem hodnoty**

Pokud tedy chceme vybrat šíp zabodnutý v zemi, použijeme označení

```
1 @e[type=minecraft:arrow, nbt= {inGround:1b}
```

## Vytvoření výbuchu

Dále budeme potřebovat vytvořit výbuch. Nejjednodušší způsob je *spawnout* **aktivované TNT**. Na to můžeme použít příkaz `/summon`

### Příkaz `summon`

```
1 /summon <entita> [<pozice>] [<nbt>]
```

- `entita`
  - ID entity, kterou chceme *spawnout*
- `pozice`
  - volitelné
  - pozice, kde chceme entitu *spawnout*
- `nbt`
  - `nbt` data, které má mít entita nastavené

Například pro aktivované TNT můžeme použít příkaz `/summon minecraft:tnt`

Pokud bychom chtěli zároveň nastavit i délku výbuchu ( `fuse` ), tak už musíme uvést i souřadnice a `nbt` data

```
1 /summon minecraft:tnt ~ ~ ~ {fuse:20}
```

### Příkaz `kill`

Poslední příkaz, který budeme potřebovat je příkaz `/kill`. Tento příkaz používáme na zničení (zabití) entity.

```
1 /kill [<cíl>]
```

- `cíl`
  - entita, kterou chceme zničit
  - pokud neuvedeme žádnou entitu, tak příkaz zabije hráče, který příkaz spustil

## Příklad

Na samotné výbušný šíp budeme potřebovat 2 příkazy.

1. příkaz pomocí kterého *spawneme* aktivované TNT na místě šípu, který je zapíchnutý v zemi (má vlastnost `inGround` s hodnotou `1b`)
2. příkaz, který zničí šíp zabodnutý v zemi
  - Pokud bychom šíp nezničili, tak se budou TNT *spawnovat* dokud šíp nezmizí

## Řešení

Použijeme 2 příkazové bloky:

Na detekci šípu v zemi a následné *spawnutí* TNT použijeme příkaz `execute`, pomocí kterého nastavíme spuštění příkazu na souřadnice všech šípů, které mají `nbt` s vlastností `inGround` a hodnotou `1b`. Na souřadnicích těchto šípů *spawneme* aktivované TNT s dobou výbuchu `0`.

```
1 /execute at @e[type=minercraft:arrow,nbt={inGround:1b}] run summon
   minecraft:tnt ~ ~ ~ {fuse:0}
```

- příkazový blok nastavíme na `opakovací` a klidně můžeme použít `vždy aktivní`

Na zničení šípu použijeme příkaz `kill`, kde opět vybereme šíp s vlastností `inGround` a hodnotou `1b`

```
1 /kill @e[type=minercraft:arrow,nbt={inGround:1b}]
```

- příkazový blok nastavíme na `řetězový`, `podmíněný` a `vždy aktivní`
  - bude se tedy spouštět pouze pokud se úspěšně provede příkaz v prvním příkazovém bloku