# IT doupě

# Jak na… Pohyb kamery v Unity

Pokud jste už někdy dělali v Unity, určitě jste potřebovali najít nějaký skript pro ovládání pohybu hráče nebo kamery. Ze svých zkušeností vím, že spousta z nich nikdy pořádně nefungovala a pokud ano, dělaly něco úplně jiného, než jsem chtěl. Proto jsem se kvůli svému vlastnímu projektu rozhodl takový skript napsat a v tomto čísle vám ukážu, jak ho můžete implementovat taky a nebo si konečně vytvořit svůj vlastní.

Snažil jsem se o to, aby kamera fungovala co nejpodobněji té v editoru, takže pomocí WASD se hýbe, kolečkem myši se jezdí nahoru a dolů a pokud podržíte pravé tlačítko myši, můžete se otáčet. Otevřete si nový Unity projekt a pokud je prázdný, ideálně do něj přidejte nějaký objekt, abyste si ověřili, že nestojíte na místě. Na kameru přidejte rigidbody, v **Constraints** zmrazte všechny osy rotace (aby se nepřekulila), hmotnost nastavte na 1x10-3 (čím větší tím pomalejší) a tření na 10 (čím menší tím déle se hýbe). Pokud nechcete, aby kamera létala skrz terén, změňte detekci kolizí na **Continuos Dynamic**.

Teď teprve začíná sranda, bude potřeba spousta vektorů. Ale není to nic těžkého, vektory jsou vlastně jen směr, kterým se bude objekt posouvat. Jeden si hned založíme, na X bude, zda jsme zmáčkli W nebo S (1 nebo -1) a na Z to samé, ale s A a D. Celé to vynásobíme rychlostí kamery, to samotné by ale k pohybu nestačilo, jezdili bychom totiž jen po hlavních osách světa. Abychom se pohybovali tím směrem, kterým se díváme, vytvoříme ještě jednu proměnou obsahující rotaci kamery. To vše nakonec přidáme pomocí quaternionu (není v mých silách vysvětlit, jak to funguje, stačí vědět, že to popisuje rotaci objektu), který hlavní osy otočí do úhlu, kterým se objekt dívá, vynásobíme to směrem a rychlostí a uděláme z toho sílu, kterou kameru odpinknem.

Doufám, že jste ještě neusnuli, tohle byly jenom 3 řádky kódu. Další na řadě je zoom, který bude kamerou hýbat nahoru a dolů. Tentokrát použijeme **AddRelativeForce()**, který veškeré úhly nastaví za nás a nám tak stačí jen vynásobit vektor reprezentující směr vpřed časem mezi aktuálním a předešlým snímkem, rychlostí zoomu a samozřejmě tím, jak moc se posunulo kolečko myši. Pokud bychom tuto metodu použili u pohybu kamery, sice bychom nemuseli řešit žádnou rotaci, pak by se ale místo po rovině objekt pohyboval v prostoru a to by hráčům docela ztížilo ovládání.

Poslední nám zbývá otáčení kamery. Tady je nejtěžší zapamatovat si, která osa myši značí kterou souřadnici – osa X jde vertikálně, osa Y horizontálně a aby toho nebylo málo, je invertovaná, takže ji je potřeba znegovat. Pak už jen vezmeme rotaci kamery a přičteme k ní, o kolik jsme ji otočili.

Teď už jen stačí nacpat všechny tři části do Update() metody a relativně jednoduchý script na ovládání kamery je na světě. Pokud jste něčemu nerozuměli, nemusíte se tím trápit – když jsem kód poprvé před rokem psal, trvalo mi 3 dny než jsem ho zprovoznil a ani teď pro mě nebylo jednoduché zjistit, co to vlastně dělá. To je pro tento měsíc všechno a jak říká pan profesor Baránek: hlavně experimentujte.

Jan Dlabaja, 2L

//img1