# **IT doupě**

**Jak na… Debug Minecraft pluginu**

Vítejte u dalšího dílu všemi oblíbené Java série. Od první části uplynulo již několik měsíců, a pokud si správně pamatujete, ukazoval jsem vám, jak plugin zkompilovat a spustit (stačilo ho přetáhnout z target do plugins složky). Pokud jste plugin sestavili jednou nebo dvakrát, nebyl v tom žádný problém. Pokud na něm ale aktivně pracujete, pak je to neefektivní a zdržující. I na to ale naštěstí existuje řešení, zabudované přímo v IntelliJ, které vám ukážu.

Možná jste si všimli, že vedle tlačítka pro spuštění projektu je takový rozklikávací obdélníček. K čemu že je? Nachází se v něm konfigurace pro export a spuštění pluginu. Když ho rozkliknete a kliknete na **Edit configuration**, otevře se okno s konfigurací. Po klepnutí na + v levém horním rohu by na vás měl vyskočit seznam všemožných konfigurací, pro naše účely ale bude stačit obyčejný jar archiv (ikona zip složky a název **JAR Application).** Po vybrání by se vpravo mělo ukázat několik nových kolonek. Nás ale bude zajímat jen pár z nich: první je **Path to JAR**. Sem zadejte cestu k .jar souboru serveru, například Spigotu nebo Paperu. K tomu se váže i kolonka **Working Directory**, kam patří cesta k celé složce se serverem. Pokud nechcete, aby vám GUI serveru spáslo celou ramku, připište do argumentů **nogui**. V JRE kolonce si můžete nastavit verzi Javy, kterou použijete. Všechny důležité kolonky jsou již sice vyplněné, to nejdůležitější nám ale přece jenom zbývá – nastavit co konkrétně chceme zkompilovat.

K tomu slouží v Intellij takzvané „artefakty“. Dá se v nich nastavit, zda chceme exportovat celý projekt, nebo jen některé knihovny. Jejich vytvoření je poměrně snadné – Ve **File -> Project Settings -> Artifacts** klikněte v levém sloupečku na + a zvolte **JAR -> Empty**, což vpravo vygeneruje pár pro nás nepodstatných kolonek a dva velmi podstatné sloupečky. V tom s nápisem Available Elements jsou všechny knihovny a části projektu, které se dají exportovat, což je v našem případě jen *compile output*. Ten stačí vzít a přetáhnout do sloupce vedle. Teď, když je artefakt připravený k použití, je třeba ho ještě přiřadit. Všimli jste si naspodu debug konfigurace takového okýnka s nápisem **Before launch**? Ne? Tak přesně do tohodle okýnka artefakt přijde, u + zvolíme **Build Artifacts** a máme vyřešeno. Toť vše, po spuštění projektu naskočí celý server i s nejaktuálnější verzí pluginu.

Oproti předchozímu debugování to má kromě mnohonásobně urychleného testování ještě jednu výhodu - HotSwap, neboli změnu kódu za běhu. To znamená, že pokud zrovna nepřidáváte novou metodu nebo třídu, můžete jednoduchým kliknutím na kladívko aplikovat všechny změny v reálném čase bez toho, abyste museli server po každém printu restartovat.

//img1, 2