

03 AKVÁRIUM – PRACOVNÝ LIST

1. Skúmaním projektu **A03_Akvarium.aia** (jeho časti **Designer** aj časti **Blocks**) odpovedzte na otázky. Ak ani po **dlhšom skúmaní** neviete odpoveď, napíšte **Neviem**.

Otázka	Odpoveď
<p>Z akých komponentov sa skladá program?</p> <p>Preskúmajte časť Components v obrazovke Designer. Napíšte mená komponentov a ich typy (typ zistíte podľa ikony tak, že nájdete rovnakú ikonu v palete komponentov) a slovne vyjadrite čo tento komponent reprezentuje.</p>	
<p>Ktorý z komponentov má v sebe obrázok pozadia akvária?</p>	
<p>Aká vlastnosť a ktorého komponentu spôsobuje, že aj keď otočíte mobil na výšku, tak sa akvárium neotočí, ale zostane stále zobrazené na šírku obrazovky?</p>	
<p>Aké vlastnosti (ktorých komponentov) alebo ktorý blok programu spôsobuje, že sa ryby hýbu? Zmeňte rýchlosť pohybu niektorej ryby.</p>	
<p>Vysvetlite slovami blok programu pre udalosť Ryba1.EdgeReached, ktorá sa hýbe „správne“.</p> <p>Čo je to Ryba1.Heading? Kedy je Ryba1.Heading = 0? Na akú hodnotu sa zmení Ryba1.Heading potom, ako sa ryba odrazí od pravého okraja? Najprv si stanovte hypotézu o hodnote Ryba1.Heading a potom si ju overte. <u>Tip:</u> V Designer si všimnite si nastavenie Heading u Ryba3.</p>	
<p>Udalosť Ryba2.EdgeReached a Ryba3.EdgeReached má rovnaký kód, ale napriek tomu sa tieto dve ryby odrážajú každá inak. Aká vlastnosť to spôsobuje?</p>	

2. Opravte projekt tak, aby sa **Ryba2** aj **Ryba3** odrážali od okrajov akvária rovnako, ako **Ryba1**.

Tip 1: Právý klik na blok ukáže menu, keď v ňom vyberieme **Duplicate**, získame kópiu bloku, v ktorej môžeme upraviť názvy rýb a pripojiť ju na iné miesto v programe.

Tip 2: Budete potrebovať súbor **ryba3_Right.png**.

3. Preskúmajte bloky, ktoré ponúka komponent **Ryba1** (rovnaké ponúkajú aj komponenty **Ryba2** a **Ryba3**). Zamerajte sa na udalosti. Vyplňte v tabuľke k názvu udalosti to, kedy udalosť vznikne. Prvý riadok je už vyplnený. Pomáhajte si svojou znalosťou angličtiny a čítajte aj popisy blokov, ktoré sa objavia, keď nad nimi necháte kurzor myši.

EdgeReached	Keď pri svojom pohybe narazí na okraj grafickej plochy (Canvas).
Dragged	
TouchDown	
TouchUp	
ColidedWith	
NoLongerColidingWith	

4. Vyriešte niektoré z nasledujúcich úloh
 - a) Aby sa rybičky pohybovali rôznou rýchlosťou.
 - b) Aby sa pri dotyku **Ryba1** otočila, akoby narazila na okraj akvária.

Tip: Blok pre vetvenie

V skupine **Control** nájdeme dva bloky začínajúce slovom **if**. Prvý z nich je príkaz **if-then**, ktorý zatiaľ nemá vetvu **else**. Druhý má vetvu **else**, ale zároveň má aj výstupok vľavo, teda dáva výsledok. Nie je to preto príkaz, ale **podmienený výraz**.

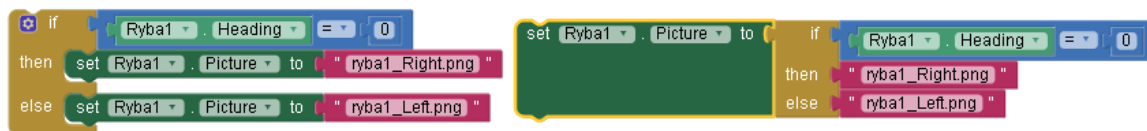


Blok pre vetvenie typu **if-then-else** v ponuke blokov App Inventora nenájdeme v žiadnej skupine blokov, musíme ho **vytvoriť**. A to tak, že zo zoznamu **Control** vyberieme prvý z blokov **if-then** a keď ho umiestnime do programu (v oblasti **Viewer** obrazovky **Blocks**), tak tam pridáme **else** použitím jeho **mutátora** – modrej ikony s ozubeným kolieskom v ľavom hornom rohu bloku:



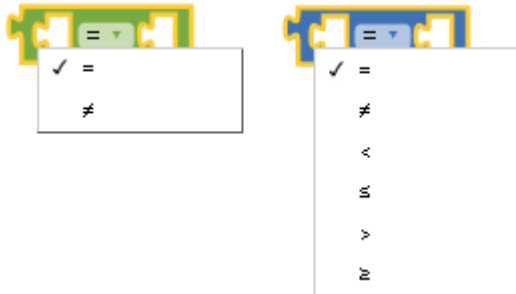
Všimnite si, že v okienku, ktoré otvorí **mutátor**, je okrem **else** na výber aj **else if**. Ten sa zide v prípade viacnásobného vetvenia. Pomocou **else if** z neho môžeme vytvoriť napríklad blok, ktorý je vidíme na obrázku vyššie vpravo.

Podmienený výraz if-then-else môžeme niekedy použiť na skrátenie programu (vyjadrenie toho istého postupu menším počtom blokov). Napríklad tieto dva kúsky programu robia to isté:



Tip: Blok pre porovnanie

Blok pre nerovnosť < alebo > priamo v zoznamoch blokov **nenájdeme**. Treba zobrať blok pre „rovná sa“ a zmeniť jeho operátor na iný. V zoznamoch blokov nájdeme dva rôzne bloky „rovná sa“. Jeden zelený v skupine **Logic** a jeden modrý v skupine **Math**. Keď klikneme na znamienko v oboch z nich vidíme, že v zelenom bloku sa nám ponúka len „rovná sa“ a „nerovná sa“, zatiaľ čo v modrom bloku nájdeme všetkých šesť relačných operátorov:



- c) Aby pri dotyku **Ryba2** zrýchliła o nejakú hodnotu, ale ak už ide veľmi rýchlo, tak aby spomalila na najmenšiu rýchlosť, prípadne, aby **Ryba2** pri dotykoch menila náhodne rýchlosť v nejakom intervale.
- d) Aby sa pri dotyku **Ryba3** zastavila a pri ďalšom dotyku, aby sa zase pohla.
- e) Aby sa dala **Ryba3** „odniesť“ na iné miesto akvária pomocou ťahania prstom (udalosť **Dragged**).
- f) Aby sa každá ryba pri zrážke s inou rybou otočila, prípadne aby pritom vydala zvuk. (Aby sme mohli testovať zrážku, musíme buď naprogramovať aj bod e) alebo zmeniť y-súradnicu niektorej ryby v obrazovke **Designer** tak, aby boli dve ryby približne v rovnakej hĺbke a mali tak možnosť "zraziť sa". Použite napr. zvuk <http://soundbible.com/1005-Frying-Pan-Impact.html>.