|  |  |  |  |
| --- | --- | --- | --- |
| Idő | Óra menete | Leírás | Comment |
| 0-5 | Köszönés, ismétlés | Gépek bekapcsolása.  Mi volt múlt órán? | Válasz: Változók létrehozása |
| 6-10 | Ismétlés | **Feladat**: Hozzunk létre egy i változót majd 0tól növeljük 1-ig az értékét.  Hozzanak létre egy int változót 0 kezdőértékkel.  i = 0;  Növeld 7ig az értékét egyesével majd írasd ki. (mindegy melyik)  i++; i += 1; i = i+1; i++; i++; i++; i++;  Erre találták ki a ciklusokat, amik ugyan azt képesek sokszor megcsinálni. Nem kell sokszor leírnunk ugyan azt, így átláthatóbb is lesz a kódunk, illetve nekünk sem kell annyit írni. |  |
| 10-15 | Do While ciklus | do{                  i++;                  Console.WriteLine(i);              }while(i!=7);  Ez a ciklus addig fog menni, amíg i nem egyenlő 7-el. Hiába teljesül elsőre is a feltétel, akkor is 1szer le fog futni. Ezért hátul tesztelő ciklusnak hívják. |  |
| 15-20 | Ellenőrzött bekérés | **Feladat**: A következő egy ellenőrzött bekérés lesz, ahol az alma szót akarjuk bekérni.  Hozzanak létre egy üres string változót beker névvel.  string beker = "";  Írassák ki, hogy „Írd be hogy alma”.  Console.WriteLine("Ird be, hogy \"alma\":");  Amit a console ablakba írunk, azt a Console.ReadLine() függvénnyel tudjuk beolvasni. Ezzel egy egész sort tudunk beolvasni.   do{                  bekert = Console.ReadLine();              }while(bekert != "alma" ); |  |
| 20-30 | If elágazás | A felhasználóbarátabb kezelés miatt írassuk ki, hogy „Próbáld újra”, így tudni fogjuk miért villog megint a kurzor.  Ehhez kell megint az i változó, amit le kell 0-zni, és a do while ciklusban egy if. Ha az i nagyobb mint 0 (minimum egyszer lefutott a kódunk) akkor kiírja, hogy „Próbáld újra!”. A végén azért írjuk ki a szót, amit bekértünk.  **Végleges**:  string beker = "";  i = 0;  Console.WriteLine("Ird be, hogy \"alma\":");  do{    if(i > 0){      Console.WriteLine("Próbáld újra!");    }    beker = Console.ReadLine(); i++;  }while(beker != "alma" );  Console.WriteLine("A szo: " + beker); |  |
| 30-55 | Órai ismétlés | **Feladat**: Írasd ki, hogy „3-as szorzótábla:”. Csináljanak egy ciklust, ami visszaadja a 3-as szorzótáblát 3-tól 30-ig. Majd kérdezze meg, hogy jól írta-e ki és várja a választ, ha igen(y) a válasz lépjen ki, de ha nem(n), akkor írja ki megint és elé, hogy „Újra próbálom!”.  i = 0; //nem szükséges  int j = 0;  beker = "";  Console.WriteLine("3-as szorzótábla:");  do{    i = 0;    if(j > 0){     Console.WriteLine("Újra próbálom.");    }  do{    i += 3;    Console.WriteLine(i);  }while(i != 30);  j++;  Console.WriteLine("Jól írtam ki?");  beker = Console.ReadLine();  }while(beker != "y"); | Ha látod, hogy nagyon elakadtak, akkor együtt csináljátok meg, de akár önálló feladat is lehet |
| 55-60 | Elköszönés | Mentsék el a projektet, nyugodtan vigyék haza (GitHub).  Gépek kikapcsolása.  Pozitív értékelés! + Jutalom: CUKORKA  Elköszönés | Cukorka, matrica csak abban az esetben jár, ha megérdemlik! |

## Az otthoni gyakorló feladatok a [3.hazi.cs](https://github.com/DrCode17/CS) file-ban elérhető