

Feladatok

1. Írj programot, amely a felhasználótól karakterláncokat kér be (a bekért karakterláncok számát is a felhasználó adhatta meg)! Ezután a program keresse meg és írja ki a képernyőre ezek közül a lexikografikusan legkisebbet!

```
public static void main(String[] args)
{
    Scanner in=new Scanner(System.in);
    System.out.println("Add meg a tömb méretét: ");
    int aSize=in.nextInt();
    String[] str=new String[aSize];
    System.out.println("\nAdj meg "+aSize+" karakterláncot: ");
    for(int i=0; i<str.length; i++)
        str[i]=in.next();

    int index=0;
    for(int i=1; i<str.length; i++)
    {
        if(str[i].compareTo(str[index])<0)
            index=i;
    }
    System.out.println("\nLexikografikusan a legkisebb kifejezés a(z) "+str[index]+".");
}
```

2. Írj programot, amely bekér a felhasználótól egy mondatot, majd kiírja, hányszor szerepel abban a „java” szó!

```
public static void main(String[] args)
{
    Scanner in=new Scanner(System.in);
    System.out.println("Kérem a mondatot: ");
    String str=in.nextLine();

    int index=0, db=0;
    while(str.indexOf("java",index)>=0)
    {
        index=str.indexOf("java",index);
        index++;
        db++;
    }
    System.out.println("\nA mondatban a 'java' kifejezés előfordulásainak a száma: "+db);
}
```

3. Írj programot, amely bekér egy karakterláncot a felhasználótól, és eldönti róla, hogy az palindrom-e!
Segítség: palindromnak nevezzük azokat a szavakat vagy mondatokat, melyek visszafelé olvasva önmagukat adják, pl.: apa, görög, abba.

```
public static void main(String[] args)
{
    Scanner in=new Scanner(System.in);
    System.out.println("Kérem a karakterláncot: ");
    String str=in.nextLine();
    boolean palindrom=true;
    for(int i=0; i<str.length()/2 && palindrom; i++)
        palindrom=(str.charAt(i)==str.charAt(str.length()-i-1));

    if(palindrom)
        System.out.println("\nPalindrom!");
    else
        System.out.println("\nNem palindrom!");
}
```

4. Írj programot, amely bekér egy mondatot, és az abban szereplő „az” szó összes előfordulását lecseréli „asd”-ra!

```
public static void main(String[] args)
{
    Scanner in=new Scanner(System.in);
    System.out.println("Kérem a mondatot: ");
    String str=in.nextLine();
    String newStr=str.replaceAll("az", "asd");
    System.out.println("\n"+newStr);
}
```

5. Készíts programot, amely bekér egy karakterláncot a felhasználótól, és az abban szereplő valamennyi magánhangzót nagybetűsre cseréli!

```
public static void main(String[] args)
{
    Scanner in=new Scanner(System.in);
    System.out.println("Kérem a karakterláncot: ");
    String str=in.nextLine();
    StringBuffer sb=new StringBuffer(str);

    for(int i=0; i<sb.length(); i++)
    {
        char ch=str.charAt(i);
        if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u')
            sb.setCharAt(i, Character.toUpperCase(ch));
    }
    System.out.println("\nÁtalakítás után: "+sb);
}
```

6. Készíts programot, amely bekér egy mondatot, és törli belőle az összes space-t!

```
public static void main(String[] args)
{
    Scanner in=new Scanner(System.in);
    System.out.print("Kérem a mondatot: ");
    String str=in.nextLine();
    StringBuffer sb=new StringBuffer(str);

    int delIndex=0;
    for(int i=0; i<str.length(); i++)
    {
        char ch=str.charAt(i);
        if(Character.isSpaceChar(str.charAt(i)))
        {
            sb.deleteCharAt(i-delIndex);
            delIndex++;
        }
    }
    System.out.print("\nA mondat szóközők nélkül: "+sb);
}
```