IDEA 1: array

setLength(INT)

create new array of INT length

set each index of array to a space “ “

return this array

changer(array)

the value of the last index will be added by 1, changing it to the next ASCII character

IDEA 2: 2d array

[reset, reset, reset, reset, reset, reset]

Symbols: 33-47, 58-64, 91-96, 123-126 (32)  
numbers: 48-57 (10)  
lowercase: 97-122 (26)  
uppercase: 65-90 (26)  
reset row (1)

Initializes 2d array

Int rowNumber = if symbols add 32, if numbers add 10, if lowercase add 26, if uppercase add 26, always add 1 for reset row

int[][] myTwoDeeArray = [passwordLength][rowNumber]  
int currentRow = 0

if symbols (these must be reordered)  
for i=1 until i>15  
for k=0 until k>passwordLength{  
myArray[currentRow][k] = i + 32;  
}  
currentRow + 1  
}

if numbers  
for i=1 until i>10  
for k=0 until k>passwordLength{  
myArray[currentRow][k] = i + 47  
}  
currentRow += 1

if symbols  
for i=1 until i>7  
for k=0 until k>passwordLength{  
myArray[currentRow][k] = i + 57;  
}  
currentRow + 1  
}

if uppercase  
for i=1 until i>26  
for k=0 until k>passwordLength{  
myArray[currentRow][k] = I + 64  
}  
currentRow + 1  
}

if symbols  
for i=1 until i>6  
for k=0 until k>passwordLength{  
myArray[currentRow][k] = i + 90;  
row +=1  
yada yada

if lowercase  
for i=1 until i>4  
for k=0 until k>passwordLength{  
myArray[currentRow][k] = i + 96;

if symbols  
for i=1 until i>4  
for k=0 until k>passwordLength{  
myArray[currentRow][k] = i + 122;  
row +=1  
yada yada

mandatory  
for k=1 until k>passwordLength  
myArray[currentRow][k] = 0;  
  
initializes 1d array  
int myOneDeeArray[] = new array(passwordLength);  
for k=0 until k>passwordLength  
1darray[k] = 2darray[0][k]

int rowOfRightmost = 0;

Brute forces, iterates  
While (currentPassword != correctPassword) {  
1darray[passwordLength-1] = [2darray][rowOfRightmost];  
rowOfRIghtmost += 1;

//checks if one of the values of 1d array is 0, indicating that it must be reset  
for k=1 until k>passwordLength  
if 1darray[k] == 0  
for i=k, i++, until i>passwordLength  
{1darray[k-1] = 2darray[0][k]  
  
//prints  
for k=1 until k>passwordLength  
system.out.print(1darray[k])

System.out.print(\n)