

 <b>bbs.eins.mainz</b> Berufsbildende Schule Technik	<b>1. Kursarbeit</b> <b>Informationsverarbeitung</b>	Name:	Stick:
		Datum:	Wesc
BGY 17	___ von ___ Punkten erreicht: ___ %	<b>Note:</b>	

### Allgemeines

- Die Programmieraufgaben speichern Sie auf dem ihnen zugewiesenen Stick im entsprechenden workspace  
(\\[Eclipse][Photon][64B]\\Stick**XX**)
- Die Lösungen müssen mit dokumentenechten Stiften (**Kugelschreiber** oder **Fine-Liner**) (keine rote Mine) erstellt werden.
- Die Bewertung des Tests ist nur bei **gut lesbarer Schrift** möglich.
- **Zugelassene Hilfsmittel:** Cheat-Sheet (im Anhang der Klausur)

### Aufgabe 1

/ 4 x 1 = 4 Pkt.

Betrachten Sie die nachfolgenden Programmausschnitt. Kreuzen Sie jeweils die entstehende Bildschirmausgabe an (nur eine Lösung ist korrekt).

(a)

```
int counter = 3;
while (counter < 7) {
    System.out.print(counter + " ");
    counter = counter + 1;
}
System.out.println();
```

- ☐ a: 3 4 5 6
- ☐ b: 2 3 4 5 6
- ☐ c: 3 + 4 + 5 + 6
- ☐ d: 3 4 5 6 7

(b)

```
int wert = 0;
for (int i = 0; i < 5; i++){
    wert = wert + i*2;
}
System.out.println("Der Wert ist: "
    + wert);
```

- ☐ a: Der Wert ist: 10
- ☐ b: Der Wert ist: 15
- ☐ c: Der Wert ist: 20
- ☐ d: Der Wert ist: 30

(c)

```
int zufallszahl = 14;
if (zufallszahl == 7) {
    System.out.print("Ihre Zahl ist ");
}
else {
    System.out.print("Ihre Zahl ist
    nicht ");
}
System.out.println("7!");
```

- ☐ a: 7!
- ☐ b: Ihre Zahl ist nicht
- ☐ c: Ihre Zahl ist 7!
- ☐ d: Ihre Zahl ist nicht 7!

(d)

```
int grenze = 14;
int summe = 0;
int zaehler = 1;
do{
    summe = summe + zaehler;
    zaehler++;
} while (summe < grenze);
System.out.println(summe);
```

☐ a: 14

☐ b: 15

☐ c: 17

☐ d: 13

## Aufgabe 2

/ 5 x 2 = 10 Pkt.

Geben Sie für die nachfolgenden Werte jeweils zwei sinnvolle Datentypen an, die Sie zur Speicherung verwenden können.

Hinweis: Sie dürfen den Datentyp *String* maximal in zwei Fällen verwenden.

(a) 3,1415

(b) Ä

(c) 1024

(d) true

(e)  $2^{32} - 123456789$


## Aufgabe 3

/ 6 + 5 + 5 + 3 = 19 Pkt.

Geben Sie für jede Datenstruktur ein kurzes aber vollständiges Beispiel!

Dieses kann so abstrakt aber auch so konkret sein, wie Sie es gestalten wollen! Es sollte aber alle relevanten Bestandteile der Datenstrukturen enthalten.

Verwenden Sie wenn nötig ein separates Blatt Papier.

(a) Bedingte Anweisung (mindestens zwei unabhängige Bedingung)

(b) Zählschleife

---

---

(c) Wiederholungsanweisung (mit Endbedingung)

---

---

(d) Wiederholungsanweisung (mit Anfangsbedingung)

## Programmieraufgaben

### Aufgabe 4

/ 1 + 4 + 8 + 2 = 15 Pkt.

Schreiben Sie ein Programm, das folgendes realisiert:

Lesen Sie zunächst vom Benutzer eine ganzzahlige **Obergrenze** ein.

Anschließend werden die ungeraden Zahlen **bis zur** angegebenen **Obergrenze** addiert.

Das Programm soll **die Summe** sowie die **Anzahl der Summanden** ausgeben.

Beantworten Sie zunächst folgende Frage:

Welche Datenstrukturen müssen Sie verwenden? (1)

---

**Beispielausgabe:** Geben Sie eine Obergrenze an: 789  
Die Summe 784 ergibt sich als Summe aus 28 ungeraden Zahlen.

### Aufgabe 5

/ 1 + 3 + 9 + 2 = 15 Pkt.

Entwerfen Sie ein Programm, dass vom Benutzer die Eingabe einer ganzen Zahl **n** fordert. Im Anschluss werden die Zahlen, beginnend bei **1 bis einschließlich** der vom Benutzer definierten Zahl **n** zusammengerechnet. Dabei muss folgende Bedingung erfüllt werden:

Ist die Zahl eine...:

- ... gerade Zahl, so wird diese zur Endsumme addiert
- ... ungeraden Zahl, so wird sie von der Endsumme subtrahiert

Das Programm gibt die **Rechnung**, sowie das **Ergebnis** aus.

Beantworten Sie zunächst folgende Frage:

Welche Datenstrukturen müssen Sie verwenden? (1)

---

**Beispielausgabe:** Geben Sie die Anzahl der zu verwendenden Elemente an: 12  
Rechnet man - 1 + 2 - 3 + 4 - 5 + 6 - 7 + 8 - 9 + 10 - 11 + 12 erhält man 6

*Hinweis: Um zu prüfen, ob eine Zahl gerade bzw. ungerade ist, empfiehlt sich die Verwendung des Modulo-Operators %.*

Viel Erfolg!

# Cheat-Sheet

## How to: Eclipse

- Starten Sie **Eclipse Photon** vom Stick!
- Bestätigen Sie beim Start die vorgeschlagene Workbench: *[Eclipse]/[Photon]/[64]\StickXX*
- Erstellen Sie ein neues Projekt:
  - File -> New -> (Java) Project
  - Wählen Sie Java Project aus und klicken Sie auf **Next**
  - Benennen Sie ihr Projekt mit ihrem Namen (Name)
  - Klicken Sie auf Finish
- Führen Sie einen Rechtsklick auf das erstellte Projekt auf und wählen anschließend: *New -> Class* aus
- Geben Sie der Klasse einen aussagekräftigen Namen gefolgt von der Aufgabennummer (z.B.: Summe04).
- Setzen Sie einen Haken im Abschnitt *Which method stubs would you like to create?* im Kästchen von **public static void main(String[] args)**.
- Klicken Sie auf Finish

## Aufbau - Klassen, Methoden, Variablen

### Klassen

Für eine Klasse gilt folgende Struktur:

```
(<Zugriffsart>) class <Bezeichner> (extends <Oberklasse>) {...}
```

Im Allgemeinen benötigen Sie lediglich **public class** Klassenname

### Methoden

Auch Methoden folgen einer gewissen Regel bezüglich des Aufbaus. Hier gilt:

```
(<Zugriffsart>) <Rueckgabewert> <Bezeichner> (<Parameter>) {...}
```

Für diese Klausur benötigen Sie lediglich folgenden Aufbau **public void** Methodenname ()

### Variablen

Auch wenn Sie Variablen verwenden müssen Sie sich an gewisse Richtlinien halten:

```
<Datentyp> variablenname = <Wert>;
```

### Scanner

Möchten Sie eine Nutzereingabe abfragen, so müssen Sie zunächst den Scanner erzeugen, um anschließend eine Nutzereingabe abzufragen:

```
Scanner eingabe = new Scanner(System.in);  
int a = eingabe.nextInt();
```