# LU02.A02 - Codeabdeckung

Definiere 10 Testfälle um eine möglichst hohe Anweisungs- und Bedingungsabdeckung zu erreichen.

## Auftrag

Du musst das unten gezeigte Programm testen. Mit genau 10 Testfällen soll eine möglichst hohe Anweisungs- und Bedingungsabdeckung erreicht werden. Erstelle dazu ein Dokument mit den Testfällen und den beiden Kennzahlen zur Abdeckung.

Um die Abdeckung zu ermitteln, markierst du für deine Testfälle:

- Welche Anweisungen (Befehle) ausgeführt wurden
- Welche Bedingungen mit `true` und welche mit `false` ausgewertet wurden.

## Sourcecode

```
def main():
    bills = read_bills()
    bills.sort(reverse=True)
    withdrawl(bills)

def read_bills():
    values = [1]
    print('Gib die Notenwerte ein')
    index = 1
    while index < 5:
        try:
            value = int(input(f'Wert der {index}. Note > '))
            values.append(value)
            index += 1
        except ValueError:
            print('Gib eine Ganzzahl ein')

    return values


def withdrawl(bills):
    amount = -1
    while amount != 0:
        try:
            amount = int(input('Betrag eingeben (0=Beenden) > '))
            if amount > 0:
                output_bills(amount, bills)
```

```
            elif amount < 0:
                print('Negative Beträge sind unmöglich')
        except ValueError:
            print('Gib eine Ganzzahl ein')


def output_bills(amount, bills):
    index = 0
    while amount > 0:
        if amount >= bills[index]:
            print(f'{bills[index]} K$')
            amount -= bills[index]
        else:
            index += 1


if __name__ == '__main__':
    main()
```

## Abgabe

Exportiere (Drucke) dein Dokument als PDF-Datei und gib diese hier in der Moodle-Aufgabe ab.

---

M450-LU01

Marcel Suter

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
**https://wiki.bzz.ch/modul/m450/learningunits/lu02/aufgaben/coverage**

Last update: **2024/03/28 14:07**