

# Übung 3

Planen und Implementieren Sie eine Web-Anwendung die aus einer Datenbank Produkte ausliest und nach Kategorien ausgibt. Die Anwendung bietet die Möglichkeit Listen von Produktkategorien und Produkten auszugeben. Als Grundlage dient eine vorhandene Datenbank (Diese finden Sie auf Moodle).

In der DB finden Sie unter anderen zwei Tabellen, welche die Produkte (products) und die Produktkategorien (product\_types) enthalten, die über einen Fremdschlüssel ( [von products] id\_product\_types → id [von product\_types] ) verbunden sind.

Die Queries die Sie benötigen sind die beiden, wobei die php Variable natürlich einen gültigen Wert enthalten muss (die ID zur gewählten Kategorie):

```
-- für die Liste der Kategorien
SELECT id, name FROM product_types ORDER BY name;

-- für die Produkte einer Kategorie
SELECT t.name AS productTypeName, p.name AS productName
FROM product_types t
JOIN products p ON t.id = p.id_product_types
WHERE t.id = {$productId};
```

## Schnittstelle

Folgende zwei Parameter sollen über die Schnittstelle verarbeitet werden:

- GET Parameter String: action (Kann die Werte **listTypes** und **listProductsById** enthalten)
- GET Parameter Integer: typeId (Im Fall listProductsById, enthält dieser Parameter die ID der gewählten Kategorie)

2 Beispiele dazu:

<http://localhost/Uebung3/index.php?action=listTypes>

<http://localhost/Uebung3/index.php?action=listProductsById&typeId=2>

## Output zur Aktion: listTypes:

Als Beispiel:

```
[ {
    "productType": "day creme",
    "url": "http://localhost/Uebung3/index.php?action=listProductsById&typeId=1"
}, {
    "productType": "night creme",
    "url": "http://localhost/Uebung3/index.php?action=listProductsById&typeId=2"
}, ... und so weiter
]
```

## Output zur Aktion listProductsByTypeld mit typeld:

Als Beispiel:

```
{
  "productType": "day creme",
  "products": [{
    "name": "Daycreme"
  }, {
    "name": "ZZ SENSITIVE - Protective Day Cream"
  }, {
    "name": "Mega-Mushroom Skin Relief - Soothing Face Cream"
  },
  ... und so weiter
  ],
  "url": "http://localhost/Uebung3/index.php?action=listTypes"
}
```

## ToDo's:

- Verwenden Sie möglichst die im Unterricht vorgestellte MVC Architektur
- Überlegen Sie wo die DB eingebunden werden kann und wie Sie dazu die Konfiguration und Datenweitergabe organisieren
- Stellen Sie, bevor Sie mit der Umsetzung starten, sicher, dass die DB verbunden wird

## Abgabe der Übung 3

- bis 17.05.2020
- im Moodle unter Uebung 3
- NACHNAME\_Uebung3.zip (kein \*.rar)
  - mit allen Files des Projektes
  - bitte keine versteckten Files die sonst in Ihrem Filesystem vorkommen \*.ts, \*.ps, usw.
  - Ihr Entwurfskonzept, wie üblich als Scan/Foto/etc.
- Fragen bitte im Diskussionsforum (Keine Scheu! Wir sind hier um zu lernen!)
- versuchen Sie es wirklich alleine
- vergleichen Sie mit der Anwendung aus Uebung1 und Uebung2
- versuchen Sie clean zu Implementieren

## Planen der zweiten Anwendung

- überlegen Sie welche Objekte wahrscheinlich eine Rolle spielen werden
- wie könnten die Namen der Objekte lauten
- welche Eigenschaften und Methoden werden die Objekte brauchen
- wo wird es notwendig sein Schleifen oder Fallunterscheidungen einzuplanen
- welche Parameter müssen weiter gereicht werden?
- zeichnen Sie einen leserlichen Plan auf