

*변수는 괄호 안으로 표시

login.php

1. 클라이언트에서 사용자의 아이디(id. String)와 비밀번호(pw. String)를 넘겨준다.
2. 해당 아이디가 존재하지 않거나, 비밀번호가 틀리면
-> return FALSE (boolean)
3. 로그인 성공 시, 다음 형태의 JSON 양식을 리턴한다.

```
userData: {  
  month: 4,  
  income: 2000000,  
  balance: 581300,  
  expense: 1418700,  
  expenseTypeCash: 1418700,  
  expenseTypeCard: 0  
},  
monthData: {  
  thisYear: 2020,  
  thisMonth: 4,  
  spendContent: {  
    1: [2000000, "+"],  
    3: [7900, "-"],  
    5: [400000, "-"],  
    7: [56800, "-"],  
    8: [6000, "-"],  
    9: [25000, "-"],  
    10: [500000, "-"],  
    12: [36700, "-"],  
    13: [259200, "-"],  
    14: [10500, "-"],  
    15: [53600, "-"],  
    16: [10200, "-"],  
    17: [5000, "-"],  
    20: [12800, "-"],  
    21: [35000, "-"]  
  }  
},
```

여기서, php의 DateTime 클래스를 이용해 '올해(thisYear. int)'와 '이번달(thisMonth. int)'에 해당되는 정보를 최초로 보냅니다.

getUserData.php

로그인과 비슷하나, 클라이언트에서 넘겨주는 변수가 없고, 유저데이터를 JSON으로 받기만 한다. 화면을 업데이트할 일이 있을 때 쓴다.

getMoneyDetail.php

1. 클라이언트에서 해당년도(thisYear. int)와 해당월(thisMonth. int), 해당일(today. int) 변수를 전달합니다.
2. 해당 날짜에 해당하는 정보를 다음 JSON양식으로 보낸다.

```
moneyDetail: {  
  moneyDetailsNum: 4,  
  moneyDetailTagData: ["경조사", "카페", "교통비", "기타"],  
  moneyDetailTagIcon: ["mdi-account-group", "mdi-coffee", "mdi-bus", "mdi-minus"],  
  iconColor: ["#FFEB3B", "#00838F", "#E57373", "#F44336"],  
  moneyDetailContentData: ["생일", "스타벅스", "택시", "에어팟"],  
  moneyDetailMoneyData: [  
    [80000, "+"],  
    [4500, "-"],  
    [4700, "-"],  
    [170000, "-"]  
  ]  
}
```

각 변수 설명

moneyDetailsNum: 그 날짜에 들어있는 정보의 수

moneyDetailTagData: 태그 정보

moneyDetailTagIcon:

아이콘 이름. <https://materialdesignicons.com/> 접속해 확인해 볼 것.

iconColor: 각 아이콘을 둘러쌀 원의 배경라운드 색깔

moneyDetailContentData: 상세내역

moneyDetailMoneyData: 금액. +면 수익, -면 지출

각 라인은 각 배열변수의 인덱스의 일치에 따른다.

ex) 1번라인 -> moneyDetailTagData[0], moneyDetailTagIcon[0] ...

2번라인 -> moneyDetailTagData[1], moneyDetailTagIcon[1] ...

setMoneyDetailExpense.php

1. 클라이언트에서 해당년도(thisYear. int)와 해당월(thisMonth. int), 해당일(today. int) 변수를 전달하고, 추가로 다음 변수를 보낸다.

ㄱ. money(int): 쓴 돈

ㄴ. content(String): 상세내역

ㄷ. icon(String): 아이콘내용 ex) mdi-account-group

2. 해당 내역을 추가하고, getMoneyDetail.php에서 리턴했던 것처럼 수정된 내역을 리턴한다.

modifyMoneyDetailExpense.php

1. 클라이언트에서 해당년도(thisYear. int)와 해당월(thisMonth. int), 해당일(today. int) 변수를 전달하고, 추가로 다음 변수를 보낸다.

ㄱ. index(int): 수정할 리스트의 인덱스 번호

- ㄴ. money(int): 쓴 돈
 - ㄷ. content(String): 상세내역
 - ㄹ. icon(String): 아이콘내용 ex) mdi-account-group
2. 해당 내역을 수정하고, getMoneyDetail.php에서 리턴했던 것처럼 수정된 내역을 리턴한다

deleteMoneyDetailExpense.php

- 클라이언트에서 해당년도(thisYear. int)와 해당월(thisMonth. int), 해당일(today. int) 변수를 전달하고, 추가로 삭제할 index(int)를 보낸다.
- 인덱스에 해당하는 정보를 삭제 후, getMoneyDetail.php에서 리턴했던 것처럼 수정된 내역을 리턴한다.

updateUserData.php

- 유저의 정보를 수정할 때 사용한다. 클라이언트에서 다음과 같은 양식의 객체를 보낸다.

```
userData: {
  month: 4,
  income: 2000000,
  balance: 581300,
  expense: 1418700,
  expenseTypeCash: 1418700,
  expenseTypeCard: 0
},
monthData: {
  thisYear: 2020,
  thisMonth: 4,
  spendContent: {
    1: [2000000, "+"],
    3: [7900, "-"],
    5: [400000, "-"],
    7: [56800, "-"],
    8: [6000, "-"],
    9: [25000, "-"],
    10: [500000, "-"],
    12: [36700, "-"],
    13: [259200, "-"],
    14: [10500, "-"],
    15: [53600, "-"],
    16: [10200, "-"],
    17: [5000, "-"],
    20: [12800, "-"],
    21: [35000, "-"]
  }
},
```

login.php에서 받은 양식과 같으니 설명하진 않겠다. 이 데이터로 유저의 해당정보를 업데이트한다.

2. 성공시,

 -> TRUE (boolean) 리턴

 실패시,

 -> FALSE (boolean) 리턴