開発エンジニア面談時テスト

日付　　2024年2月28日

所属

お名前　UN GREFFIN NAMYU

問１. HTTPリクエストのGETメソッドとPOSTメソッドの違いを簡潔にご回答ください。

一番大きな違いはパラメーターの伝える方にあります。GETはパラメーターをURLに載せるのに対してPOSTはパラメーターをリクエストのボディに載せます。

開発エンジニア面談時テスト

問２. 会員制Webサイトでセッション・ハイジャックの脆弱性が発見され、対策を実施することになりました。なお、このWebサイトの認証及び認可の仕組みは下記の通りとなっています。

- セッションはCookieで管理されています。

- Webサイトは、非会員でも閲覧可能なページと、会員専用のページとで構成されています。

- 会員専用ページへのアクセスには、必ずログインが求められます。

- ログインの状態はセッションで管理されています。

このWebサイトでセッション・ハイジャック対策として必要となる対応を、簡潔にご回答ください。

頭に最初に浮かぶのはサーバーがセッションだけではなくて、ユーザーのIPアドレスも保存します。リクエストが来る時に来るリクエストのIPアドレスとセッションIDを存在したIPアドレスとセッションIDと比べます。合わないとセッションIDを無効にしてログインページに送ります。

アプリによってはセッションに有効期限を付けば良いこともあります。

開発エンジニア面談時テスト

問３. ２種類のIDに基づき処理を分岐するという要件に基づき、オブジェクト指向言語を用いて下記の実装が行われました。

if（ID = = "ID-A") {

// ID-A 用の複雑な処理

:

:

｝else if（ID = = "ID-B") {

// ID-B 用の複雑な処理

:

:

｝else｛

// エラー処理

:

:

}

ところがその後、仕様変更となり、新しいIDが複数件追加されることになりました。

また今後も随時、ID追加が発生する見込みです。

1）このプログラムをそのまま踏襲して改修を進めた場合、どのような問題があり得ますか。

2）このプログラムをオブジェクト指向に基づき改善する場合、どのようにすれば良いでしょうか。

それぞれ、簡潔にご回答ください。

1．このコードはハードコードと言われます。コードの匂いです。IDがどんどん増えるとそのファイルの行数が増えるし、読みにくくなるし、IDのバリューによってアプリが遅くなる可能性もあります。

2．インターフェイスや親クラスなどを作ってIDを別々のクラスにしましょう。インターフェイスを使うことでif-elseは要らなくなります。読みやすくなって、そんなに遅くなるとは思いません。同じや似ているコードがあればそれも一緒に使えるようになります。

開発エンジニア面談時テスト

問４. ある文字列が「重複する文字を含むか否か」を判断するプログラムを記述してください。

たとえばそのプログラムを使って「abcdde」という文字列を処理した場合は、「重複がある」と判定され、「abcde」という文字列を処理した場合は、「重複がない」と判定されます。

ただし対象とする文字列はASCII文字列のみとします。

また、回答として記述をおこなうプログラム言語は、実在するプログラム言語ではなく疑似言語でもかまいません。

誤解かもしれませんが、ASCII文字列のASCIIコードは32から126までように思いますので。こうなります。

import sys

def main(str):

d = {}

for c in str:

if ord(c) < 32 or ord(c) >= 127: # char -> int

continue

if (c in d): # 辞書にすでにあると重複があるのがわかる

print('重複がある')

return

else: # 辞書にないと辞書に追加する

d[c] = 1

print('重複がない')

if \_\_name\_\_ == '\_\_main\_\_':

main(sys.argv[1])

[shika@fedora Downloads]$ python test.py abcdde

重複がある

[shika@fedora Downloads]$ python test.py abcde

重複がない

[shika@fedora Downloads]$ python test.py abcdeああ

重複がない

[shika@fedora Downloads]$ python test.py abcdeああd

重複がある