

IT 基礎 2-1

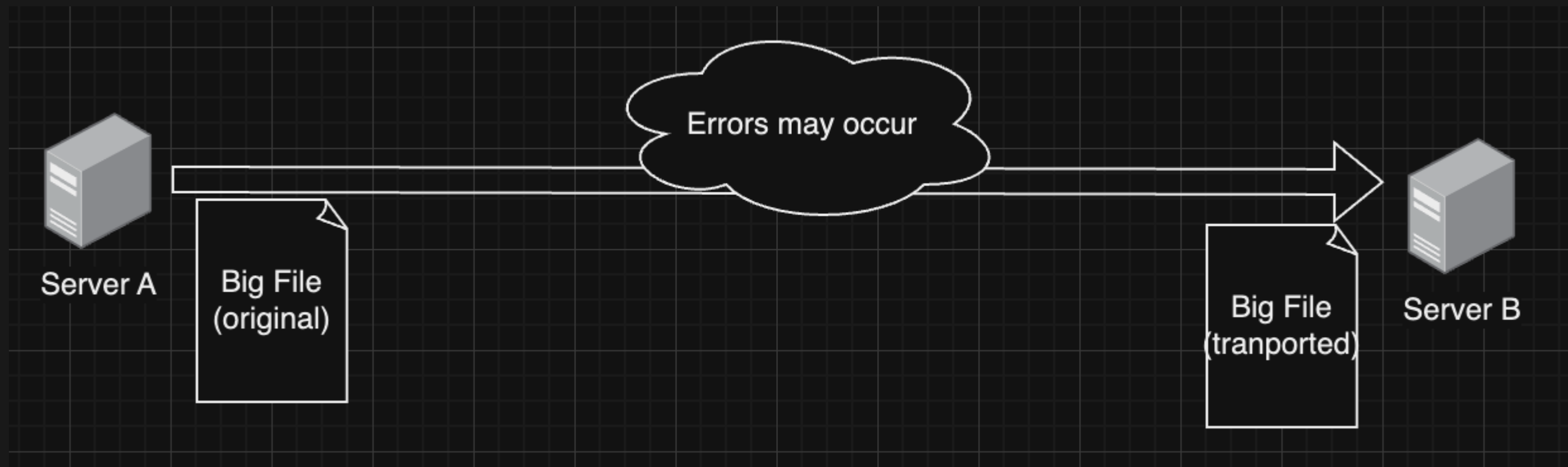


講師：王紹宇

2023-12-17

ウォーミングアップ

- ファイルの完全性（欠損や改ざんがないこと）のチェック問題
 - 前提条件
 - ネットワークが不安定で、エラー率が高い
 - ネットワーク転送のコストが高いため、転送のデータ量をなるべく最小化する
 - サーバAから巨大なファイル（数ギガ）をサーバBに転送したとする
 - 質問 - Bサーバにある転送されたファイルは、サーバAにあるオリジナルのファイルと完全一致するか、どうチェックする？



ダイジェスト(DIGEST) / 指紋 (FINGERPRINT) / 拇印 (THUMBPRINT)

ハッシュ(Hash)関数を使って確率的に完全性を保証する

→ 確率は無限に 100%に近づけることが可能

- よく使われるハッシュ関数
 - MD5 (128 bits)
 - SHA-1 (160 bits)
 - SHA-256 (256 bits)

```
$ cat result.txt
hello shell
goodbye shell
goodbye shell
$ cat result.txt | md5
c5b68aaa5e11fec99a88e7ce00d1a84f
$ cat result.txt | shasum
48fbdfef1fcef42ed4db7228c9fdf699e4f97ace0 -
$ cat result.txt | shasum -a 256
887c4cdd56bba9ee6fbf38548db0be6655add9a9259ecca7277b05ecf804d9a0 -
```

今週のコンテンツ

1-3、WEB アプリケーション基礎知識（土曜）

- ネットワーク基礎
- 各種のサービスとプロトコル
- リクエスト/レスポンス
- HTTP プロトコルの概要
- 認証/認可

2-1、開発基礎（日曜）

- プログラミング入門

豆知識へ

プログラミング入門

- コメント
- 予約語
- データ型とリテラル
- 変数と宣言
- 演算子
- 文と式
- 条件分岐
- ループと反復処理
- 配列とオブジェクト
- 関数とスコープ

コメント (COMMENT)

- JavaScript comment

```
// print a hello world message
console.log('Hello world!') // hello world

/*
This is a multi-line
comment example
*/
console.log('Goodbye!')
```

- Java comment

```
// This is a comment
System.out.println("Hello world!"); // This is a comment
/* The code below will print the words Hello World
to the screen, and it is amazing */
System.out.println("Goodbye!");
```

- Python comment

```
# comment
print("Hello world!") # another comment
```


予約語 (RESERVED WORD)

- 予約語 wiki
- JavaScript の予約語
- Java の予約語
- Python の予約語

データ型(DATA TYPE)とリテラル(LITERAL)

- [データ型 wiki](#)
- [リテラル wiki](#)
- [Javascript のデータ型とリテラル](#)
- [Java のデータ型](#)
- [Java のリテラル](#)

※以下の概念は、この Javascript の書籍を参考資料として解説

- 変数(Variable)と宣言 (Declaration)
- 演算子(Operator)
- 文(Statement)と式(Expression)
- 条件分岐(Condition)
- ループと反復処理(Loop)
- 配列(Array)とオブジェクト(Object)
- 関数(Function)とスコープ(Scope)

宿題

1. ワーク用ディレクトリを作って、その配下に cd する
2. 下記コマンドでユーザ入力を受け付けるためのライブラリをインストールする

- `npm install prompt-sync`

3. 同ディレクトリ配下で、下記 Javascript の例を `guess.js` として保存し、実行してみる

- `node guess.js`

- ```
const prompt = require('prompt-sync')()
const input = prompt('Guess a number (0-9):')
const inputNum = Number(input)
if (inputNum === 6) {
 console.log('You guessed a very lucky number!')
} else {
 console.log('You guessed', inputNum)
}
```

4. それを編集して、シンプルな Guess ゲーム([参考](#))を実現しよう

- 多分利用される文法

- 条件分岐、演算子、文と式、ループ(`continue` or `break`)

(自分なりのルールでも OK! )

- 動画閲覧 (<https://cn.chikounavi.co.jp/links/>)
  - フロントエンド開発
    - HTML、CSS、JS の役割 1
    - HTML、CSS、JS の役割 2
    - JavaScript の歴史とライブラリを解説！ フレームワークとの違いは？

# 質問応答

つづく