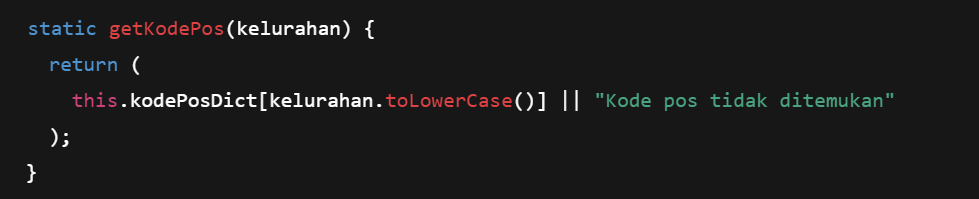
Kelas KodePos



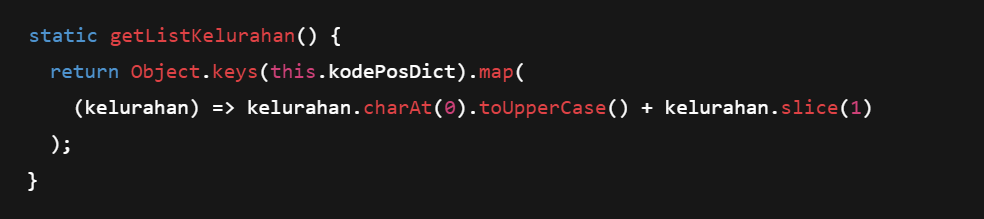
**Penjelasan:**

* **kodePosDict** adalah objek yang menyimpan data pasangan antara **kelurahan** (sebagai key) dan **kode pos** (sebagai value).
* Setiap nama kelurahan seperti "batununggal", "kujangsari", dll, memiliki kode pos terkait yang ditulis di sebelahnya, misalnya "batununggal": 40266.



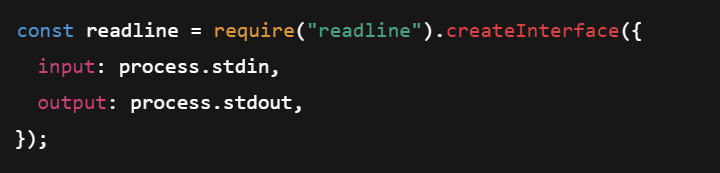
**Penjelasan:**

* Fungsi **getKodePos(kelurahan)** menerima nama kelurahan sebagai parameter.
* Fungsi ini **mencari kelurahan** dalam objek kodePosDict (dengan mengubah nama kelurahan menjadi huruf kecil menggunakan toLowerCase() agar pencarian tidak case-sensitive).
* Jika kelurahan ditemukan, **kode pos** yang sesuai akan dikembalikan.
* Jika kelurahan **tidak ditemukan**, maka pesan "Kode pos tidak ditemukan" akan dikembalikan.



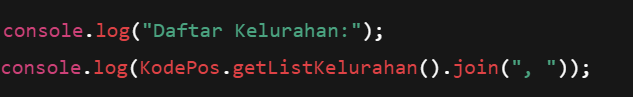
**Penjelasan:**

* Fungsi **getListKelurahan()** mengembalikan **daftar kelurahan** yang ada dalam objek kodePosDict.
* Object.keys(this.kodePosDict) menghasilkan array yang berisi nama-nama kelurahan (sebagai string).
* Fungsi map digunakan untuk memodifikasi setiap nama kelurahan, membuat huruf pertama menjadi kapital menggunakan charAt(0).toUpperCase() dan menggabungkannya dengan sisa kata menggunakan slice(1).
* Hasil akhirnya adalah daftar kelurahan dengan huruf pertama kapital.



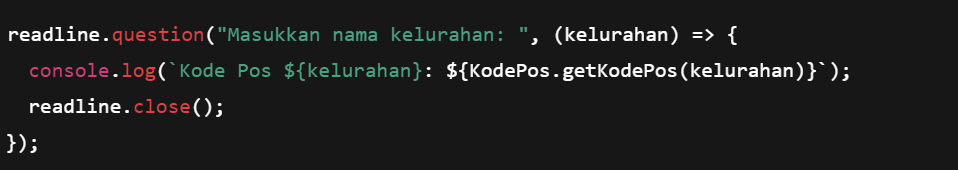
**Penjelasan:**

* Bagian ini menginisialisasi **antarmuka readline** yang memungkinkan kita untuk **menerima input** dari pengguna melalui terminal (CLI).
* input: process.stdin artinya input berasal dari terminal.
* output: process.stdout artinya output akan ditampilkan ke terminal.



**Penjelasan:**

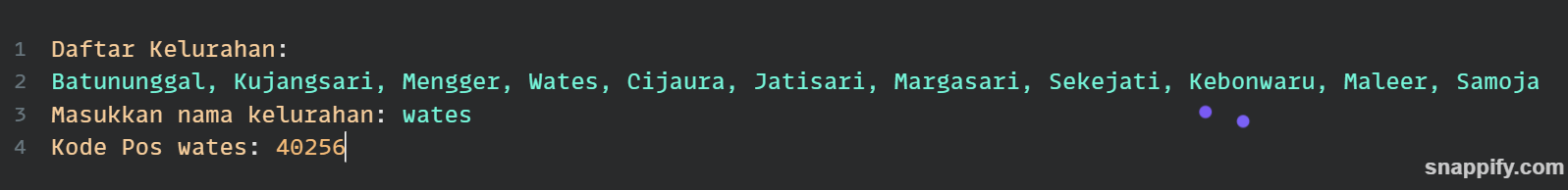
* **console.log(KodePos.getListKelurahan().join(", "))** memanggil fungsi getListKelurahan() untuk mendapatkan daftar kelurahan.
* Daftar kelurahan ini akan ditampilkan ke pengguna dalam format yang dipisahkan koma.



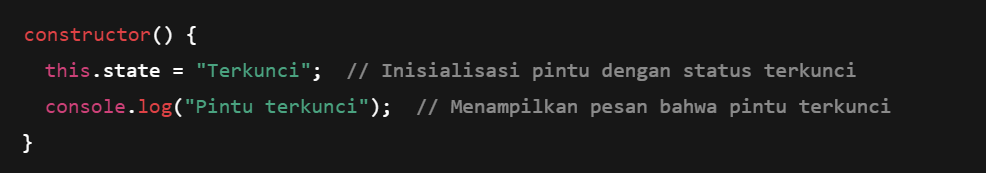
**Penjelasan:**

* **readline.question()** menampilkan pesan untuk meminta input dari pengguna. Pesan yang ditampilkan adalah **"Masukkan nama kelurahan:"**.
* Setelah pengguna memasukkan nama kelurahan, input akan diteruskan ke fungsi callback (kelurahan).
* Fungsi ini **memanggil getKodePos(kelurahan)** untuk mencari kode pos kelurahan yang dimasukkan oleh pengguna.
* Hasilnya akan dicetak dengan pesan seperti **Kode Pos wates: 40256**.
* Setelah selesai, **readline.close()** dipanggil untuk menutup antarmuka input.

Output :

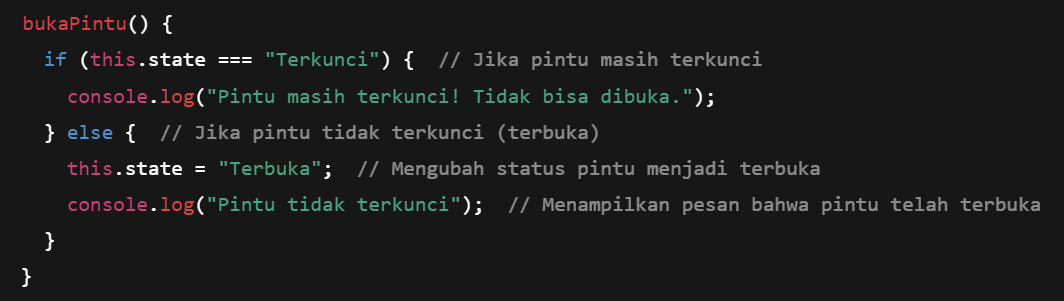


Kelas doormachine



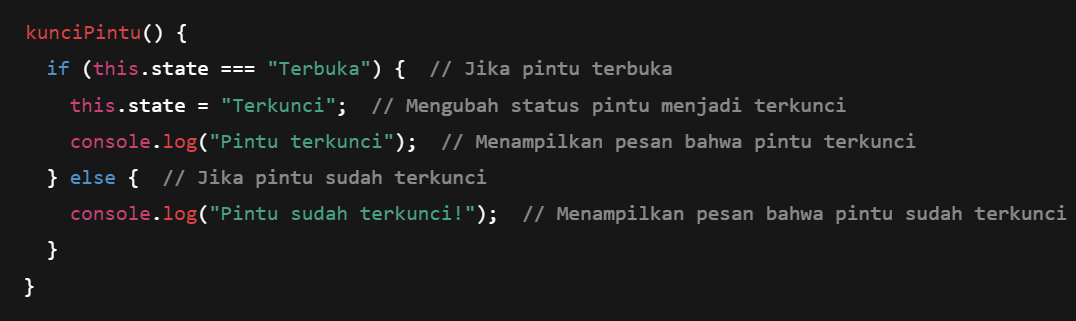
**Penjelasan:**

* **state**: Pada saat objek DoorMachine pertama kali dibuat, status pintu diatur ke **"Terkunci"**.
* **console.log("Pintu terkunci")**: Pesan ini akan muncul di konsol untuk memberi tahu bahwa pintu telah terkunci pada saat objek pertama kali dibuat.



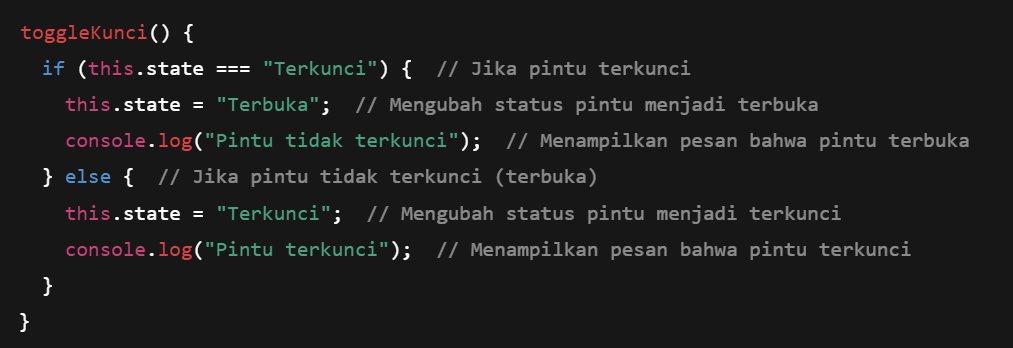
**Penjelasan:**

* **if (this.state === "Terkunci")**: Mengecek apakah pintu dalam keadaan terkunci.
* **console.log("Pintu masih terkunci! Tidak bisa dibuka.")**: Jika pintu terkunci, maka muncul pesan bahwa pintu tidak bisa dibuka.
* **this.state = "Terbuka"**: Jika pintu tidak terkunci (artinya terbuka), status pintu diubah menjadi **"Terbuka"** dan menampilkan pesan **"Pintu tidak terkunci"**.



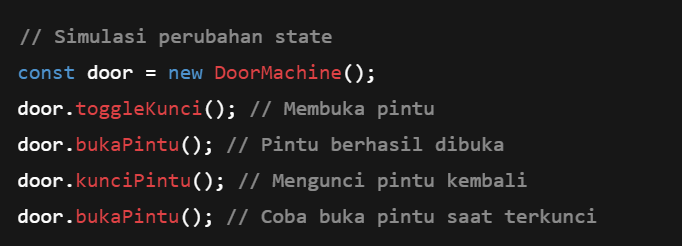
**Penjelasan:**

* **if (this.state === "Terbuka")**: Mengecek apakah pintu dalam keadaan terbuka.
* **this.state = "Terkunci"**: Jika pintu terbuka, maka pintu dikunci.
* **console.log("Pintu terkunci")**: Menampilkan pesan bahwa pintu terkunci.
* **else**: Jika pintu sudah terkunci, akan muncul pesan **"Pintu sudah terkunci!"**.



**Penjelasan:**

* **if (this.state === "Terkunci")**: Mengecek apakah pintu dalam keadaan terkunci.
* **this.state = "Terbuka"**: Jika pintu terkunci, maka status pintu diubah menjadi **"Terbuka"** dan muncul pesan **"Pintu tidak terkunci"**.
* **else**: Jika pintu sudah terbuka, maka status pintu diubah menjadi **"Terkunci"** dan muncul pesan **"Pintu terkunci"**.



**Penjelasan:**

* **const door = new DoorMachine();**: Membuat objek baru dari kelas DoorMachine. Pada saat ini, pintu pertama kali diinisialisasi dalam keadaan **"Terkunci"**.
* **door.toggleKunci();**: Mengubah status pintu menjadi **"Terbuka"** dan menampilkan pesan **"Pintu tidak terkunci"**.
* **door.bukaPintu();**: Pintu yang sudah terbuka, maka akan menampilkan pesan **"Pintu tidak terkunci"**.
* **door.kunciPintu();**: Pintu yang sebelumnya terbuka, sekarang akan dikunci kembali dan menampilkan pesan **"Pintu terkunci"**.
* **door.bukaPintu();**: Menguji coba membuka pintu yang sudah terkunci, yang seharusnya menampilkan pesan **"Pintu masih terkunci! Tidak bisa dibuka."**

Output :

