

PurpleKai — Linguagem de Programação Interpretada com Palavras-chave em Francês

Miguel Santana da Costa

Introdução

A linguagem **PurpleKai** é uma linguagem de programação interpretada, orientada a objetos, com tipagem dinâmica. Inspirada na sintaxe do C++, ela utiliza **palavras-chave em francês**, tornando a escrita mais natural para falantes do idioma. Entretanto, a documentação técnica é bilíngue, com descrições em **português e inglês** para acessibilidade e padronização internacional.

Mapeamento de Palavras-Chave: PurpleKai ↔ C++

Tabela 1: Correspondência entre palavras-chave da PurpleKai (francês) e C++ (inglês)

PurpleKai	C++	Tipo	Descrição
classe	class	Declaração	Define uma classe
hérite	: public	Herança	Estabelece herança entre classes
fonction	-	Função	Declaração de função/método
constructeur	[ClassName]	Método	Construtor da classe
self	this	Ponteiro	Autorreferência ao objeto atual
super	::	Acesso	Acesso a membros da superclasse
laisser	auto	Variável	Declara variável com tipo inferido
constant	const	Constante	Declaração de constante
si	if	Controle	Condicional simples
sinon	else	Controle	Bloco condicional alternativo
pour	for	Loop	Loop com inicialização, condição e incremento
tantque	while	Loop	Loop com condição de entrada
retourner	return	Fluxo	Retorna valor de função
nouveau	new	Memória	Alocação dinâmica de objetos
vrai/faux	true/false	Booleanos	Valores lógicos
nul	nullptr	Ponteiro	Valor nulo para ponteiros
essayer	try	Exceção	Bloco de proteção para exceções
attraper	catch	Exceção	Captura de exceções
jeter	throw	Exceção	Lançamento de exceções
interface	abstract	Interface	Classe abstrata pura
implémenter	: public	Interface	Implementação de interface

Palavras-chave da Linguagem

A tabela a seguir apresenta as palavras-chave reservadas de PurpleKai, com suas descrições em português e inglês.

Palavra-chave (FR)	Descrição (PT)	Description (EN)
classe	Declara uma classe orientada a objetos	Declares an object-oriented class
structure	Define uma estrutura simples	Defines a simple struct
public	Membro acessível publicamente	Publicly accessible member
prive	Membro privado, acessível apenas internamente	Private member, accessible only internally
protégé	Membro protegido, visível por subclasses	Protected member, visible by subclasses
si	Declaração condicional (se)	Conditional statement (if)
sinon	Bloco alternativo à condição (senão)	Alternative block (else)
tantque	Laço de repetição enquanto a condição for verdadeira	Loop while condition is true
pour	Laço com contador e condição de parada	For loop with counter and stop condition
interrompre	Interrompe um laço de repetição	Breaks a loop iteration
continuer	Avança para a próxima iteração do laço	Skips to the next loop iteration
retourner	Retorna um valor de uma função	Returns a value from a function
fonction	Declara uma função	Declares a function
nouveau	Instancia um novo objeto dinamicamente	Instantiates a new object dynamically
ceci	Referência ao próprio objeto (this)	Reference to the current object (this)
nul	Representa valor nulo ou ausência de objeto	Represents null value or no object
vrai	Valor booleano verdadeiro	Boolean value true
faux	Valor booleano falso	Boolean value false
laisser	Declara uma variável mutável	Declares a mutable variable
constant	Declara uma constante (imutável)	Declares an immutable constant
afficher	Envia saída para o console	Outputs data to console
lire	Lê entrada do usuário	Reads input from user
importer	Importa um módulo ou biblioteca	Imports a module or library

Palavra-chave (FR)	Descrição (PT)	Description (EN)
essayer	Inicia um bloco de tratamento de erro	Starts a try block for error handling
attraper	Captura uma exceção lançada	Catches a thrown exception
lancer	Lança uma nova exceção	Throws an exception
dans	Utilizado em laços do tipo "para item em lista"	Used in loops like "for item in list"
objet	Referência genérica a um objeto instanciado	General reference to an instantiated object

Exemplo Comparativo

```
// PurpleKai
classe Animal {
    fonction constructeur(n) {
        self.nom = n;
    }

    fonction parler() {
        retourner "??";
    }
}

classe Chien hérite Animal {
    fonction parler() {
        retourner "Woof!";
    }
}
```

```
// C++
class Animal {
public:
    Animal(string n) : nom(n) {}

    virtual string parler() {
        return "??";
    }
};

class Chien : public Animal {
public:
    string parler() override {
        return "Woof!";
    }
};
```

Observações Importantes

- A palavra-chave **fonction** na PurpleKai é opcional para métodos (podendo ser omitida)
- **laisser** permite redesignação de tipos (tipagem dinâmica) enquanto **auto** em C++ mantém tipagem estática
- **hérite** e **implémenter** são semanticamente equivalentes ao operador **:** do C++
- O sistema de exceções é similar, mas PurpleKai usa terminologia francesa
- PurpleKai possui tipagem dinâmica enquanto C++ é estaticamente tipado
- A linguagem oferece palavras-chave adicionais como **afficher** e **lire** para operações básicas de I/O

Conclusão

PurpleKai é uma linguagem que une a clareza do francês com a robustez da programação orientada a objetos, oferecendo uma sintaxe mais acessível para falantes de línguas latinas. Este documento estabelece a base semântica da linguagem, servindo como referência para desenvolvedores, criadores de ferramentas e educadores, com equivalências claras para C++ para facilitar a transição entre as linguagens.