* Anonymous class trong Java là một lớp không có tên, được khai báo và khởi tạo trong một lần định nghĩa. Nó cho phép định nghĩa lớp và tạo đối tượng của lớp đó mà không cần phải tạo một lớp mới hoặc triển khai một interface.
* Anonymous class thường được sử dụng để triển khai các interface hoặc lớp trừu tượng mà chỉ được sử dụng một lần. Nó cũng thường được sử dụng để triển khai các phương thức cục bộ hoặc xử lý sự kiện.
* Để tạo một anonymous class, ta sử dụng từ khóa **new** kèm theo việc triển khai lớp hoặc interface mà không cần đặt tên cho lớp đó. Sau đó, ta có thể ghi đè các phương thức của lớp hoặc interface đó để cài đặt chúng theo nhu cầu.

A picture containing text, electronics, screenshot, software

Description automatically generated

1. Functional programming

* Là một Paradigm Shift

1. Lambda Expressions

* Có một vấn đề: Nếu như triển khai của anonymous class rất đơn giản, ví dụ một interface chỉ có 1 method, thì cú pháp của anonymous class sẽ không rõ ràng. Trong TH đó, chúng ta thường truyền chức năng như một argument đến 1 method khác, như là hành động nào nên được thực hiện khi ta ấn vào nút. Biểu thức Lambda cho phép bạn làm điều này, để coi chức năng là đối số của phương thức hoặc mã là dữ liệu.
* Biểu thức lambda là một tính năng được giới thiệu trong Java 8 cho phép bạn viết mã ngắn gọn và đẹp (Trực quan) hơn bằng cách biểu thị một phương thức dưới dạng biểu thức lambda. **Biểu thức lambda là một hàm ẩn danh có thể được sử dụng để thay thế một instance của interface chỉ có một abstract method (vẫn có thể có nhiều default method và static method (được giới thiệu trong java 8)), còn được gọi là Functional Interface.**

1. Các syntax lambda
2. Single Parameters Lambdas:

* Dấu ngoặc đơn có thể được ẩn đi vì chỉ có 1 tham số.
* VD: x -> x \* x;

1. Multiple Parameters Lambdas:

* Các tham số được đóng gói ở trong ngoặc và đc ngăn cách bởi dấu phẩy
* VD: (x, y) => x + y;

1. No parameter lambdas:

* Cặp ngoặc rỗng
* VD: () => System.out.println(“Hello”);

1. Multi-line lambdas

* Return value

A picture containing text, screenshot, software, multimedia

Description automatically generated

1. Single-line Lambdas

* 

1. Lambdas with Explicit Types:



1. Method References

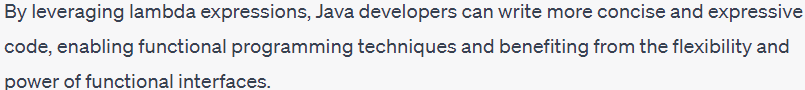
* **Thay vì cung cấp lambda body, tham chiếu đến một phương thức hiện có.**
* **Phương thức được tham chiếu đó sẽ triển khai cho functional interface (Tương tự như triển khai bằng local class hay anonymous class), nhưng cách này triển khai bằng cách tạo một method ở trong một class để triển khai.**
* Chúng ta dùng lambda expressions để tạo một anonymous class. Tuy nhiên, đôi lúc lambda không làm gì mà chỉ gọi một phương thức đang có sẵn. Trong TH này, thường sẽ rõ ràng hơn khi tham chiếu một method có sẵn bằng tên gọi của nó.
* Các loại method references:

+ Tham chiếu đến static method

+ Tham chiếu đến instance method

+ Tham chiếu đến constructor

A picture containing text, screenshot, font, document

Description automatically generated

A screenshot of a computer code

Description automatically generated with low confidence

1. Stream

* Là một chuỗi các kí tự hỗ trợ các hoạt động tổng hợp tuần tự và song song.