**Zadanie 4 Wątki-słuchacze.**

Stwórz mechanizm obsługi wyników zadań zwracanych przez wątki, działający na zasadzie słuchaczy (metoda "done" klasy FutureTask). Przetestować na przykładach.

W odpowiedzi, oprócz kodu umieść wyniki działania.

W przedstawionym kodzie, obsługa wyników zadań zwracanych przez wątki odbywa się przy użyciu nadpisanej metody done(). Metoda done() w klasie FTask<T> jest wywoływana po zakończeniu zadania, zarówno w przypadku pomyślnego zakończenia, jak i w przypadku przerwania lub anulowania zadania. Jej głównym zadaniem jest obsługa różnych scenariuszy, które mogą wystąpić po zakończeniu wykonania zadania:

* Sprawdzenie, czy wątek został przerwany:

Thread.currentThread().isInterrupted() służy do sprawdzenia, czy bieżący wątek, który wykonywał to zadanie, został przerwany (został zgłoszony sygnał przerwania). Jeśli tak, to metoda wypisuje informację o przerwaniu zadania.

* Sprawdzenie, czy zadanie zostało anulowane:

isCancelled() jest metodą dostępną w klasie FutureTask, która sprawdza, czy zadanie zostało anulowane za pomocą metody cancel(boolean mayInterruptIfRunning). Jeśli tak, metoda wypisuje informację o anulowaniu zadania.

* Pobranie wyniku zadania (jeśli nie zostało przerwane ani anulowane):

Jeśli zadanie nie zostało ani przerwane, ani anulowane, metoda próbuje pobrać wynik z zadania za pomocą this.get(). Ta operacja może rzucać wyjątki, takie jak InterruptedException lub ExecutionException, które są odpowiednio obsługiwane w bloku catch. Po uzyskaniu wyniku, metoda wypisuje informację o zakończeniu zadania wraz z wynikiem.

**Klasa FTask** rozszerzająca klasę FutureTask i zawierająca nadpisana metodę done():Obraz zawierający tekst, zrzut ekranu, Czcionka, oprogramowanie

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

**Klasa Task1** implementuje interfejs Callable<Integer> i reprezentuje zadanie zwracające wynik. Metoda call() symuluje obliczenia zajmujące czas poprzez dodawanie wartości w pętli i usypianie w każdej iteracji. Zwraca ostateczny wynik.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie

**Klasa TaskTest** jest klasą służącą do przetestowania funkcjonalności związanej z wielowątkowym wykonywaniem zadań. Tworzona jest lista ArrayList<FTask<Integer>> aL, która będzie przechowywać obiekty FTask. Wykorzystany jest obiekt ExecutorService i przy użyciu exec.execute(t), zadania są przekazywane do wykonawcy. Trzy zadania są anulowane za pomocą metody cancel(boolean mayInterruptIfRunning). Na koniec wywoływane są metody shutdown() i awaitTermination() na obiekcie exec w celu zakończenia jego pracy i oczekiwania na zakończenie wszystkich zadań w puli wątków.

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Czcionka

Opis wygenerowany automatycznie

Przykład działania:

Obraz zawierający tekst, zrzut ekranu, Czcionka, dokument

Opis wygenerowany automatycznie