

Laboratorio di Informatica

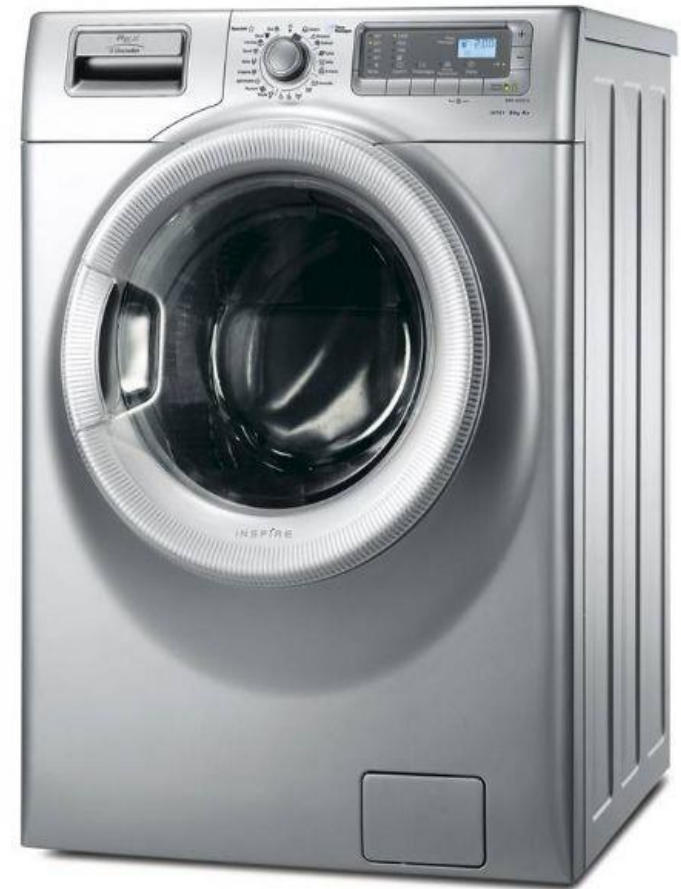
Documentazione

docente: Cataldo Musto

cataldo.musto@uniba.it

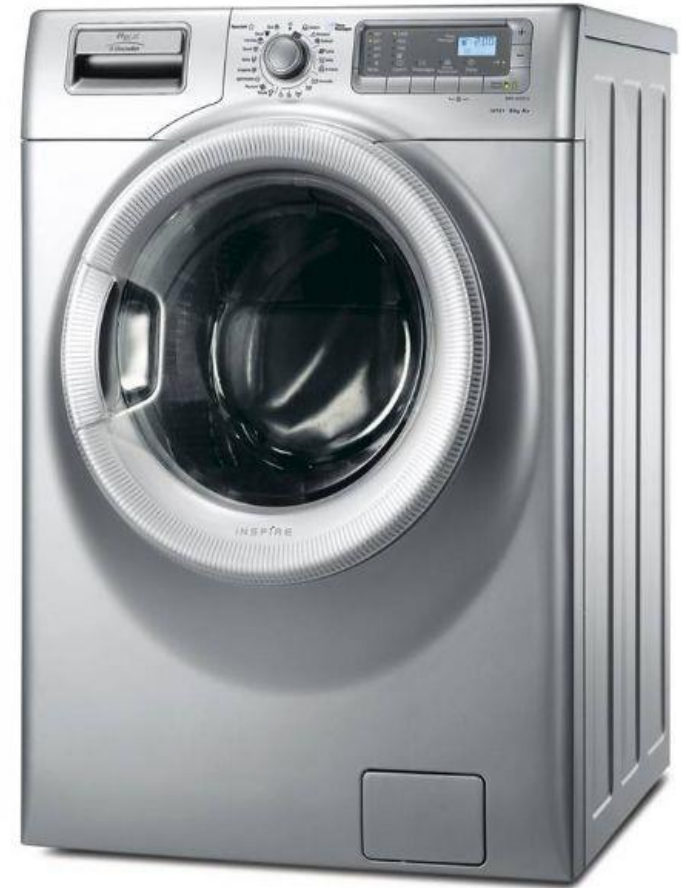
Documentare un Prodotto

- Una lavatrice è un prodotto
- **Come è documentato?**



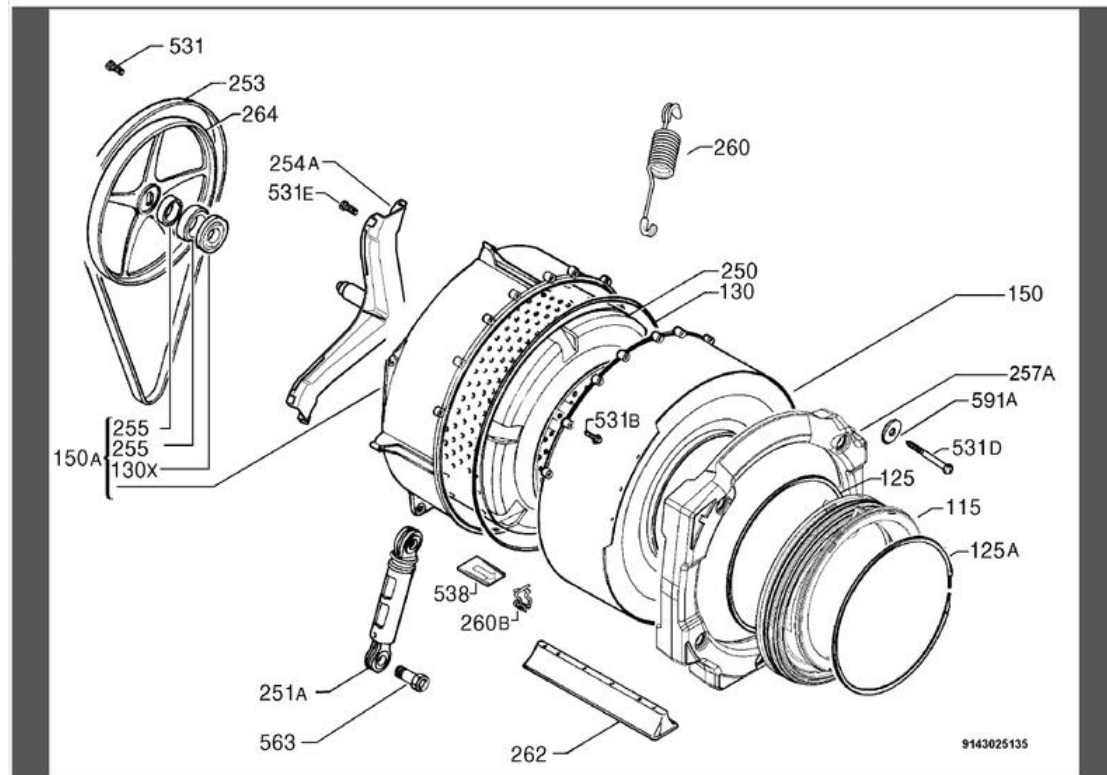
Documentare un Prodotto

- Una lavatrice è un prodotto
- **Come è documentato?**
 - Con il suo manuale tecnico e manuale d'uso



Documentare un prodotto

- **Manuale Tecnico**
 - **Chiarisce come è fatto** un prodotto



Documentare un prodotto

- Manuale d'uso
 - Chiarisce cosa fa un prodotto

SCHEDA PROGRAMMI											
<p>A. Spia "Apertura porta" </p> <p>Questa lavatrice è dotata di funzioni di sicurezza automatiche in grado di riconoscere e segnalare tempestivamente eventuali guasti e di reagire in modo adeguato, ad es. mediante le spie:</p> <p>B. Spia "Service" </p> <p>C. Spia "Ingresso acqua" </p> <p>D. Spia "Pulizia filtro" </p>						<p>E. Manopola programmi </p> <p>F. Tasto "Avvio/Pausa" </p> <p>G. Tasto "Centrifuga" </p> <p>H. Indicatore delle fasi del programma (sopra la riga)</p> <p>Spia Avvio ritardato (sotto la riga)</p> <p>I. Tasto "Annulla" </p> <p>J. Tasto "Avvio ritardato" </p>					
Programma	Simboli di lavaggio	Carico max. kg	Tipo di carico/Note			Detersivi e additivi			Opzioni selezionabili		
			- Rispettare le raccomandazioni del fabbricante riportate sull'etichetta di lavaggio			Pre-lavaggio	Lavaggio principale	Ammorbidente	Rapido	Pre-lavaggio	Risciacquo intensivo
Cotone 95 °C	95 °C	8.5	Capi bianchi in tessuto di cotone resistente, da mediamente a molto sporchi. In caso di capi molto sporchi o con macchie, si può aggiungere un additivo candeggiante a base di ossigeno che può essere utilizzato a 95 °C. Questo programma elimina i batteri, garantendo l'igienizzazione dei capi.			*	Si	*	—	*	*
Cotone 40 - 60 °C	40 - 60 °C	6.0 4.0	Biancheria da letto, tovaglie e biancheria intima, asciugamani, camicie ecc. in tessuto di cotone e lino, da mediamente a molto sporchi.			*	Si	*	*	*	*
Jeans 40 °C	40 °C	6.0 4.0	Jeans e indumenti di cotone mediamente sporchi in fibra robusta simile al jeans, ad es. pantaloni e giubbotti.			*	Si	*	*	—	*
Sintetici 30 - 40 - 60 °C	30 - 40 - 60 °C	3.0 4.0 5.0	Capi in tessuto di fibre sintetiche (come poliestere, poliacrilico, viscosa ecc.) o misto cotone, mediamente sporchi.			*	Si	*	*	*	*
Delicati 30 - 40 °C	30 - 40 °C	1.5 4.0 5.0	Tendaggi, abiti, gonne, camicie, bluse e altri capi delicati.			*	Si	*	—	—	*
Misti 40 °C	40 °C	4.0 6.0 4.0	Capi in tessuto resistente di cotone, lino, fibre sintetiche e tessuti misti, da poco a mediamente sporchi. Programma efficace della durata di un'ora. Assicurarsi di caricare solo capi di colori simili nella lavabiancheria.			—	Si	*	—	—	*
Rapido 30 °C	30 °C	3.0 4.0 5.0	Indumenti indossati per brevi periodi di tempo di cotone, poliestere, poliammide e misto cotone. Programma rapido per rinfrescare la biancheria.			—	Si	*	—	—	*
Lana Afreddo -40 °C	* -40 °C	1.0 4.0 5.0	Capi in lana, lino, viscosa o seta (rispettare le raccomandazioni sull'etichetta di lavaggio) per i quali è consigliato il lavaggio a mano o in lavatrice. Questo programma, a 40 °C, è testato e approvato da The Woolmark Company per il lavaggio in lavatrice e a mano di capi in lana con il marchio Pura Lana Vergine.			—	Si	*	—	—	*
Cotone Eco 40 - 60 °C	40 - 60 °C	6.0 4.0	Biancheria di cotone mediamente sporca. Programma più conveniente in termini di consumo combinato di acqua ed energia elettrica per il lavaggio di biancheria di cotone. - Programma di riferimento per l'etichetta dei dati energetici.			—	Si	*	—	—	*
Risciacquo & Centrifuga		—	Corrisponde all'ultimo risciacquo e alla centrifuga finale del programma "Cotone".			—	—	*	—	—	*
Centrifuga		—	Centrifuga intensiva. Corrisponde al ciclo di centrifuga del programma "Cotone".			—	—	—	—	—	*
Scarico acqua		—	Solo scarico - senza centrifuga. Metodo alternativo per terminare il programma dopo la funzione "Antipiega".			—	—	—	—	—	*

Documentare un prodotto

- Anche il software è un **prodotto**
 - Un prodotto è un insieme di attributi tangibili e intangibili di un bene o un servizio volti a procurare un beneficio a un utilizzatore, ottenuto tipicamente attraverso un processo di produzione

Documentare un prodotto

- Anche il software è un **prodotto**
 - Un prodotto è un insieme di **attributi** tangibili e intangibili di un bene o un **servizio** volti a procurare un beneficio a un **utilizzatore**, ottenuto tipicamente attraverso un processo di **produzione**
 - **attributi** = funzionalità del software
 - **servizio** = lo scopo del software
 - **utilizzatore** = utente del software
 - **produzione** = sviluppo del software

Documentare un Prodotto

- Una libreria è un modulo di un **software più grande**
- Anche una **libreria software** può essere vista **come un prodotto**
 - **attributi**: ciò che la libreria fornisce (funzioni, procedure, tipi di dati, costanti, etc.)
 - Quali metodi implementa
 - **servizio**: lo scopo della libreria
 - **utilizzatore**: sviluppatore che usa la libreria
 - **produzione**: programmazione della libreria
- Nel documentare un prodotto dobbiamo **distinguere tra interfaccia e implementazione**

Un prodotto ha un'interfaccia e un'implementazione

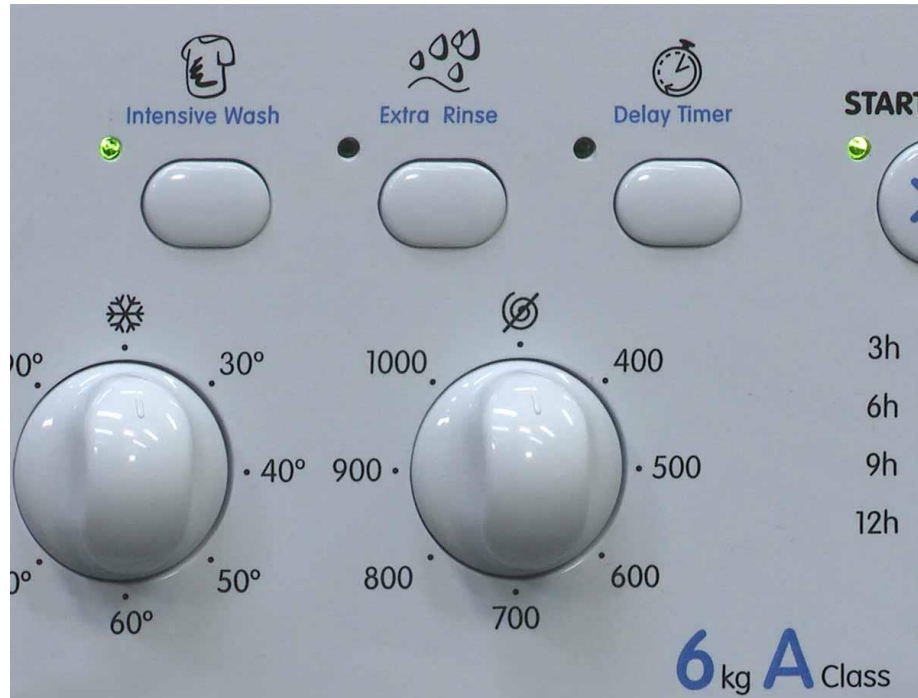


Interfaccia = Come il prodotto è «mostrato» al pubblico



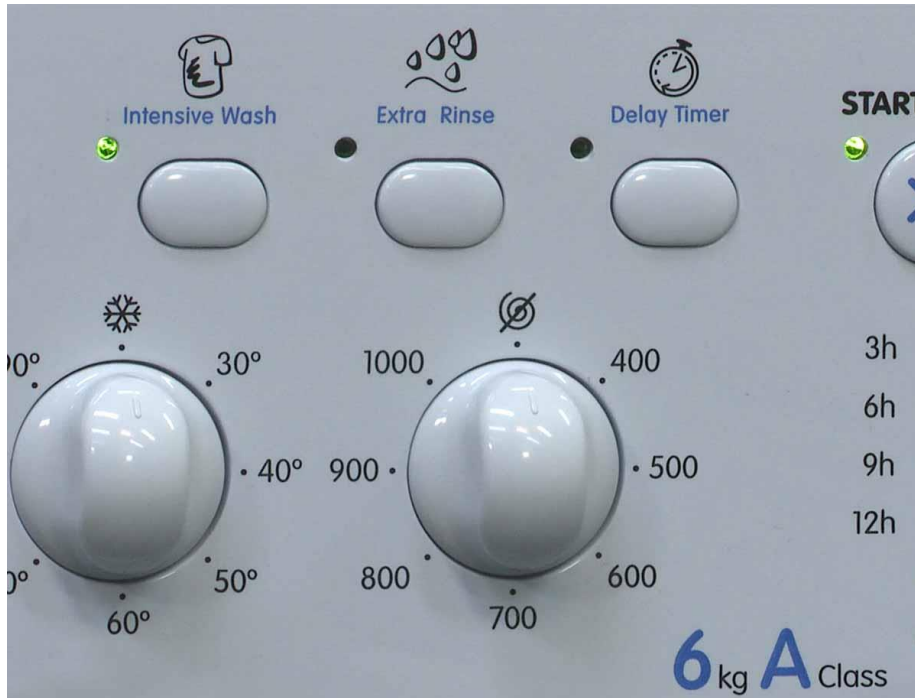
Implementazione = Come è strutturato internamente

Un prodotto ha un'interfaccia e un'implementazione



Nel nostro caso l'interfaccia sono le «funzioni» implementate in una libreria, mentre invece **l'implementazione è appunto il codice sorgente che le realizza**

Un prodotto ha un'interfaccia e un'implementazione



Interfaccia = Come il prodotto è «mostrato» al pubblico

Nel caso del software, **possono essere generate entrambe le tipologie di documentazione**

La documentazione dell'interfaccia «spiega» come la libreria o il software può essere utilizzato da un utente finale, dunque **equivale a un manuale d'uso**. Ci concentriamo su quello.

La documentazione legata all'implementazione fornisce dettagli **sul processo che ha portato alla realizzazione del prodotto** (documento di analisi, documento di progettazione, etc.), ed **equivale a un manuale tecnico**

Documentare una Libreria Software

- Qual è l'interfaccia di una **libreria software**?
 - Rappresenta **la descrizione degli attributi pubblici**, cioè delle **funzionalità utilizzabili dall'utilizzatore**
 - In C, l'interfaccia è definita dal **file header** di un modulo

Documentare una Libreria Software

- Qual è l'interfaccia di una **libreria software**?
 - Rappresenta **la descrizione degli attributi pubblici**, cioè delle **funzionalità utilizzabili dall'utilizzatore**
 - In C, l'interfaccia è definita dal **file header** di un modulo
- Fornire semplicemente il codice dell'interfaccia non è sufficiente a rendere utilizzabile una libreria
 - E' necessaria **una documentazione aggiuntiva**

Documentare una Libreria Software

- Cosa includere
 - Descrive il **significato dei tipi** e delle **costanti pubbliche**
 - Descrive **COSA fanno le funzioni** e le procedure
 - NON descrive **COME le funzioni e le procedure svolgono il loro compito**
 - → Information Hiding!

Come detto, documentare il codice equivale scrivere un **manuale d'uso**, in questo caso dell'interfaccia di una libreria

Documentazione vs. commento

- Bisogna fare una importante distinzione tra la produzione della documentazione di un prodotto software e il commento del codice sorgente
- **Documentazione**
 - Spiega il significato di una funzione **all'utilizzatore** della funzione
- **Commento**
 - Spiega i punti più critici al **programmatore** della funzione

Documentazione - Esempio

Questa è la documentazione della funzione 'atof' nel linguaggio C

Double atof(const char *str);

The string pointed to by the argument **str** is converted to a floating-point number (type **double**). Any initial whitespace characters are skipped (space, tab, carriage return, new line, vertical tab, or formfeed). The number may consist of an optional sign, a string of digits with an optional decimal character, and an optional e or E followed by a optionally signed exponent. Conversion stops when the first unrecognized character is reached.

On success the converted number is returned. If no conversion can be made, zero is returned. If the value is out of range of the type double, then HUGE_VAL is returned with the appropriate sign and ERANGE is stored in the variable errno. If the value is too small to be returned in the type double, then zero is returned and ERANGE is stored in the variable errno.

Documentazione di una funzione

Quali informazioni **inserire quando documentiamo una funzione?**

1.Cosa fa la funzione

- a. Scenari tipici
- b. Scenari eccezionali

Documentazione di una funzione

Quali informazioni **inserire quando documentiamo una funzione?**

1.Cosa fa la funzione

- a. Scenari tipici = Quale è il comportamento «normale» della funzione
- b. Scenari eccezionali = Cosa fa la funzione in caso di situazioni «anomale»

Documentazione di una funzione

Quali informazioni **inserire quando documentiamo una funzione?**

1.Cosa fa la funzione

- a. Scenari tipici = Quale è il comportamento «normale» della funzione
- b. Scenari eccezionali = Cosa fa la funzione in caso di situazioni «anomale»

2.Significato e dominio dei parametri

Documentazione di una funzione

Quali informazioni **inserire quando documentiamo una funzione?**

1.Cosa fa la funzione

- a. Scenari tipici
- b. Scenari eccezionali

2.Significato e dominio dei parametri

3.Esempi d'uso

Documentazione di una funzione

Quali informazioni **inserire quando documentiamo una funzione?**

1.Cosa fa la funzione

- a. Scenari tipici
- b. Scenari eccezionali

2.Significato e dominio dei parametri

3.Esempi d'uso

4.Pre-condizioni

5.Post-condizioni

Documentazione di una funzione

Quali informazioni **inserire quando documentiamo una funzione?**

1.Cosa fa la funzione

- a. Scenari tipici
- b. Scenari eccezionali

2.Significato e dominio dei parametri

3.Esempi d'uso

4.Pre-condizioni

5.Post-condizioni

6.Eventuali riferimenti (ad altre funzioni, link, testi, standard, etc.)

7.Eventuali note (p.e. funzioni deprecated, portabilità, etc.)

Documentazione: Pre-condizione

- **E' una condizione che deve essere vera prima** della chiamata di una funzione
 - Solitamente, è una condizione sui parametri
- Se una pre-condizione è falsa, **il funzionamento non è garantito**
 - **Esempio:** la funzione per il calcolo del BMI ha come pre-condizione che peso e altezza siano maggiori di zero ed eventualmente minori di una certa soglia
- Il chiamante di una procedura ha la responsabilità di garantire la verifica di tutte le precondizioni
 - **Una pre-condizione falsa è sintomo di un bug!**

Documentazione: Pre-condizione

- **IMPORTANTE:** la definizione delle pre-condizioni **non implica il mancato uso della programmazione difensiva!**
- E' bene specificare le pre-condizioni **ma anche rendere il codice solido** in modo tale da restituire risultati anche quando le pre-condizioni non sono verificate.

Documentazione: Pre-condizione

- **IMPORTANTE:** la definizione delle pre-condizioni **non implica il mancato uso della programmazione difensiva!**
- E' bene specificare le pre-condizioni **ma anche rendere il codice solido** in modo tale da restituire risultati anche quando le pre-condizioni non sono verificate.
- **Sapere a priori quali errori possono verificarsi è diverso dallo scrivere codice in grado di gestirli!**

Documentazione: Post-condizione

- è una condizione che **deve essere vera** dopo l'esecuzione di una **funzione**
 - Solitamente, **su parametri di output**
- La falsificazione di una post-condizione (se le pre-condizioni sono verificate) è **un chiaro sintomo della presenza di un bug nella funzione.**
 - **Esempio:** se il BMI non corrisponde a quello corretto, evidentemente c'è un **problema nell'implementazione.**

Documentazione in linea

- Oltre a documentare le funzioni nel modo convenzionale, si può documentare una libreria **nel codice sorgente stesso**
- E' necessario che tale documentazione **sia poi accessibile all'utilizzatore della libreria**
- La documentazione in linea è scritta **mediante commenti speciali**
 - Si parla di «**doc-comments**»
 - Sono commenti (dunque ignorati dal compilatore) **ma seguono una convenzione particolare**

Generazione automatica di Documentazione

- Esistono tool particolari che esaminano il codice sorgente alla ricerca di **doc-comment**
- Tutti i doc-commenti vengono formattati in documenti in formato *pdf, html, etc.*
- Si chiamano tool di **generazione automatica della documentazione**
 - Da usare per documentare il caso di studio

Generazione automatica di Documentazione

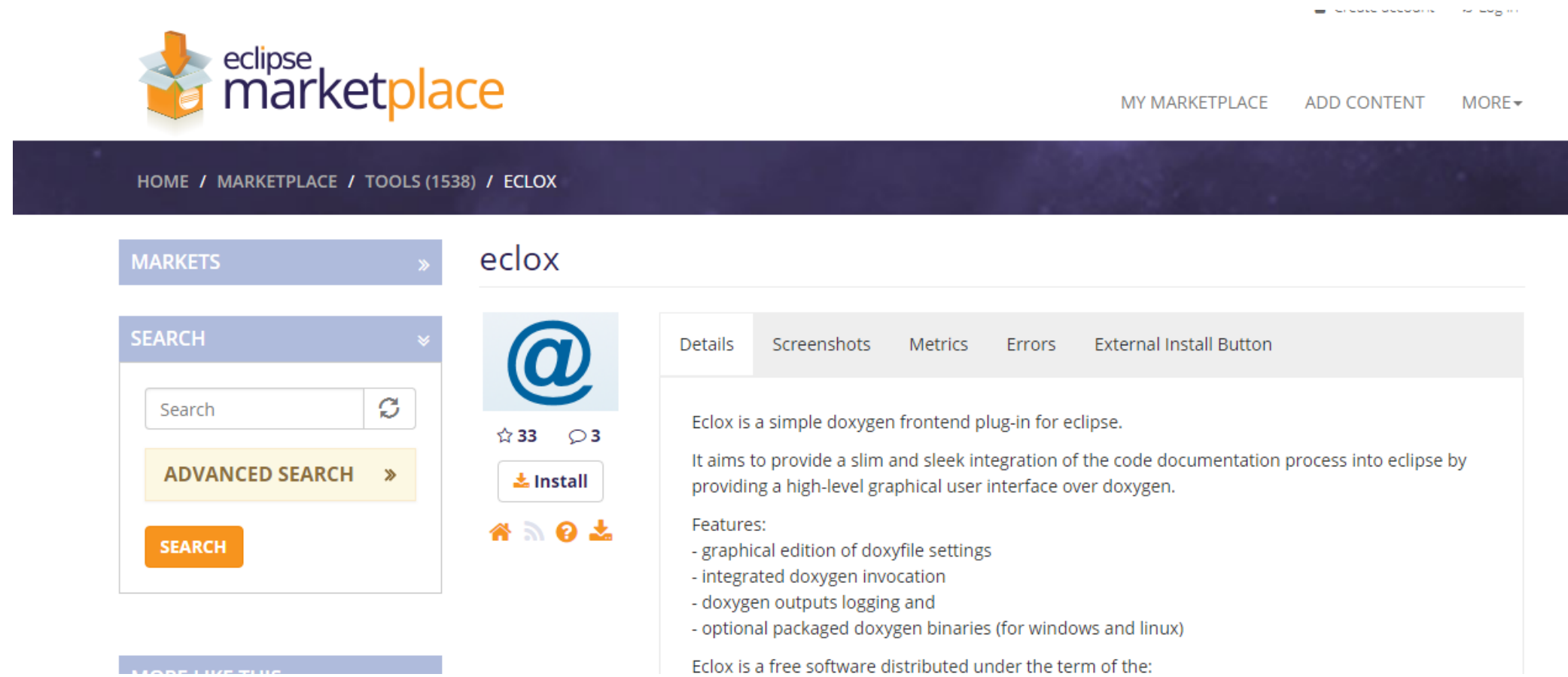
- Esistono tool particolari che esaminano il codice sorgente alla ricerca di **doc-comment**
- Tutti i doc-commenti vengono formattati in documenti in formato pdf, html, etc.
- Si chiamano tool di **generazione automatica della documentazione**
 - Da usare per documentare il caso di studio
- **Doxygen** (www.doxygen.org)
 - Multi-linguaggio, ma in particolare per C/C++
 - Si ispira a JavaDoc (per Java)
 - **Useremo le sue convenzioni per scrivere doc-comment**

Installazione Doxygen

- Per integrare Doxygen nei nostri progetti Eclipse si utilizza **Eclox**
 - Plugin Eclipse per Doxygen
 - **URL:** <http://www.stack.nl/~dimitri/doxygen/download.html>
- Per Windows, selezionare la versione (.zip) senza GUI oppure quella con interfaccia grafica
 - <http://ftp.stack.nl/pub/users/dimitri/doxygen-1.8.14.windows.x64.bin.zip>
 - Macchine a 64 bit
 - <http://ftp.stack.nl/pub/users/dimitri/doxygen-1.8.14.windows.bin.zip>
 - Macchine a 32 bit
 - <http://ftp.stack.nl/pub/users/dimitri/doxygen-1.8.14-setup.exe>
 - Versione con interfaccia grafica
 - **Scaricare il file e decomprimere il contenuto in un cartella**

Installazione Doxygen


<http://marketplace.eclipse.org/content/eclox>



The screenshot shows the Eclipse Marketplace interface. At the top, there's a navigation bar with links like 'HOME / MARKETPLACE / TOOLS (1538) / ECLOX'. The main content area is divided into two columns. The left column contains a 'MARKETS' sidebar with a search bar and an 'ADVANCED SEARCH' button. The right column displays the details for the 'eclox' plugin. The plugin's icon is a blue '@' symbol. It has 33 stars and 3 comments. Below the icon is an 'Install' button. The 'Details' tab is selected, showing a description: 'Eclox is a simple doxygen frontend plug-in for eclipse. It aims to provide a slim and sleek integration of the code documentation process into eclipse by providing a high-level graphical user interface over doxygen.' Below the description is a list of features: graphical edition of doxyfile settings, integrated doxygen invocation, doxygen outputs logging and, and optional packaged doxygen binaries for windows and linux. At the bottom, it states 'Eclox is a free software distributed under the term of the:'.

MARKETS »


SEARCH ▾

Search 


ADVANCED SEARCH »





SEARCH

eclox



☆ 33 3

 **Install**

Details Screenshots Metrics Errors External Install Button

Eclox is a simple doxygen frontend plug-in for eclipse.

It aims to provide a slim and sleek integration of the code documentation process into eclipse by providing a high-level graphical user interface over doxygen.

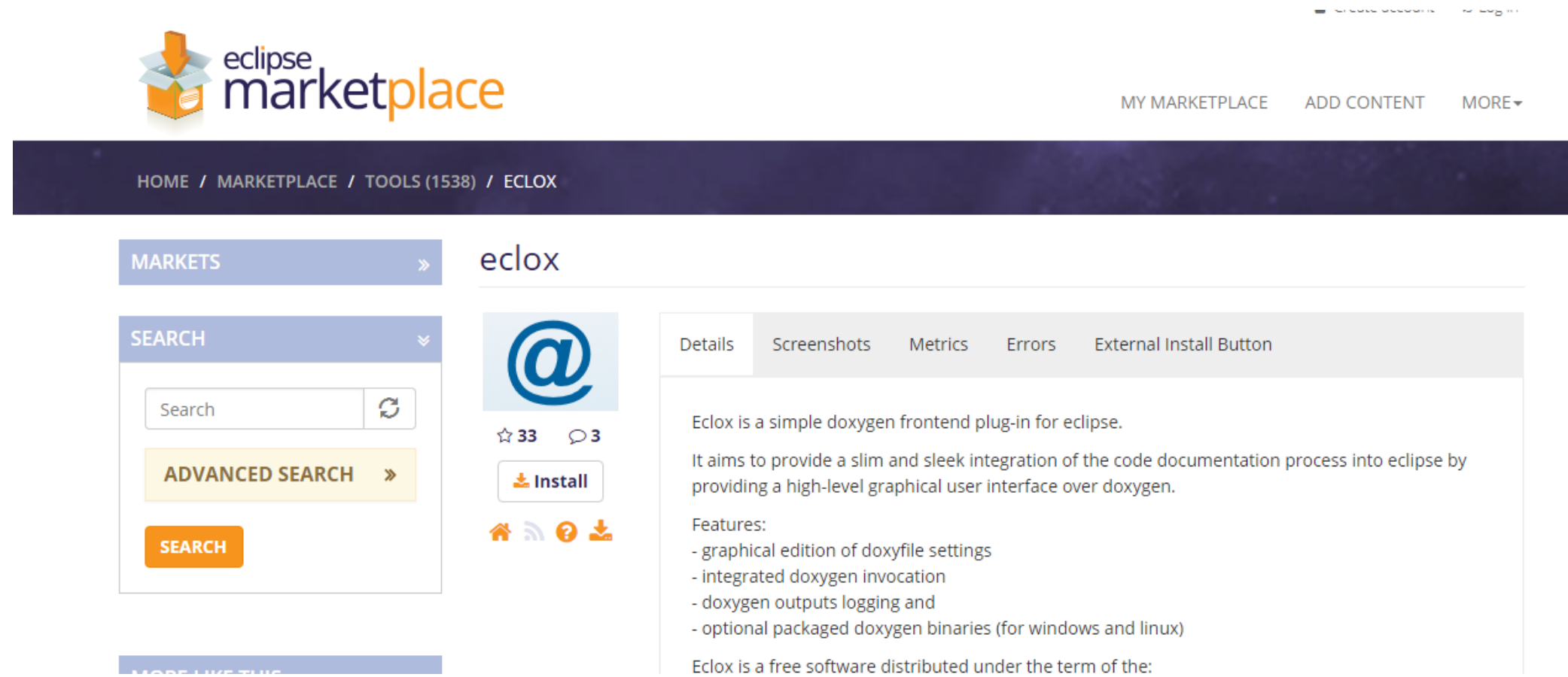
Features:

- graphical edition of doxyfile settings
- integrated doxygen invocation
- doxygen outputs logging and
- optional packaged doxygen binaries (for windows and linux)

Eclox is a free software distributed under the term of the:

Installazione Doxygen

Si clicca su **Install** e si trascina il pacchetto dentro il workspace di Eclipse



The screenshot shows the Eclipse Marketplace interface. At the top, there's a navigation bar with links: MY MARKETPLACE, ADD CONTENT, and MORE. Below this is a breadcrumb trail: HOME / MARKETPLACE / TOOLS (1538) / ECLOX. The main content area is divided into two columns. The left column contains a 'MARKETS' sidebar with a search bar and an 'ADVANCED SEARCH' button. The right column displays the 'eclox' plugin details. It features a large '@' icon, a star rating of 33, and a comment count of 3. A prominent 'Install' button is visible. Below the icon are social media and utility icons. The details section includes tabs for Details, Screenshots, Metrics, Errors, and External Install Button. The 'Details' tab is active, showing a description of 'eclox' as a simple doxygen frontend plug-in for eclipse, its purpose of integrating code documentation, a list of features (graphical edition of doxyfile settings, integrated doxygen invocation, doxygen outputs logging, and optional packaged doxygen binaries), and a note about its free software distribution.

eclox

Details Screenshots Metrics Errors External Install Button

Eclox is a simple doxygen frontend plug-in for eclipse.

It aims to provide a slim and sleek integration of the code documentation process into eclipse by providing a high-level graphical user interface over doxygen.

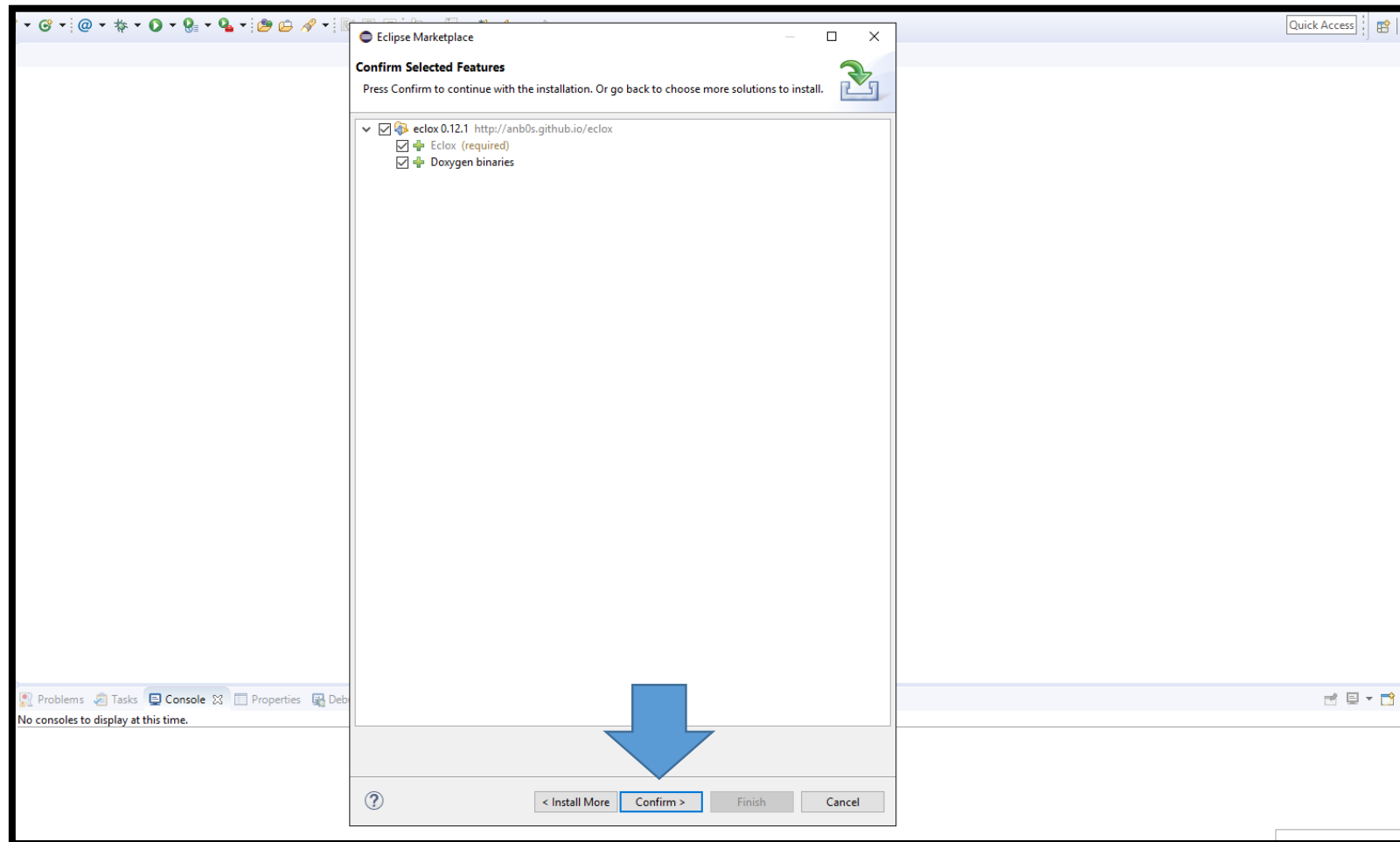
Features:

- graphical edition of doxyfile settings
- integrated doxygen invocation
- doxygen outputs logging and
- optional packaged doxygen binaries (for windows and linux)

Eclox is a free software distributed under the term of the:

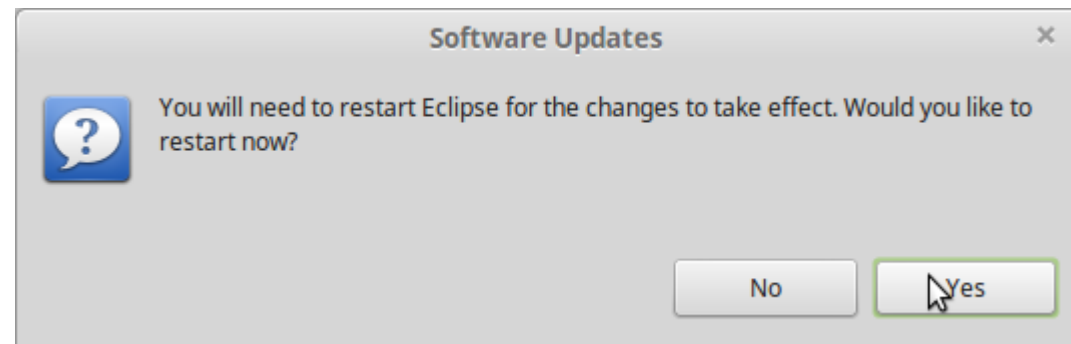
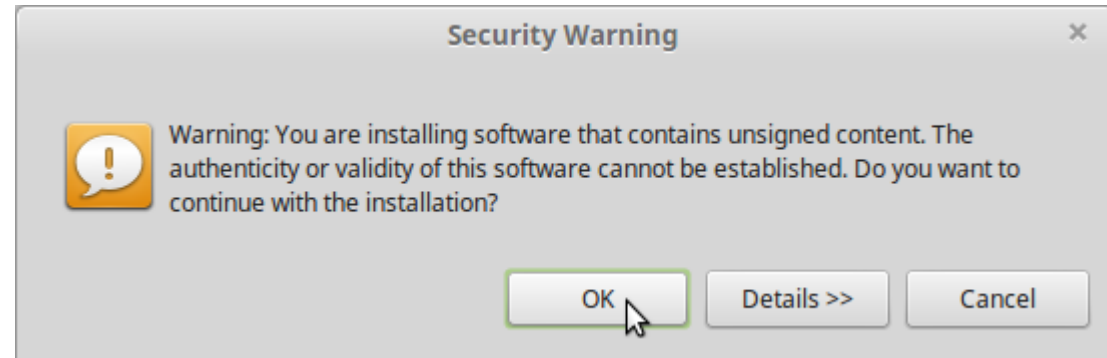
Installazione Doxygen

Cliccare su 'Confirm' e seguire la procedura di installazione guidata



Installazione Doxygen

- **Proseguire nel wizard e terminare accettando i termini della licenza**
- **Successivamente ignorare i warning e procedere al riavvio.**



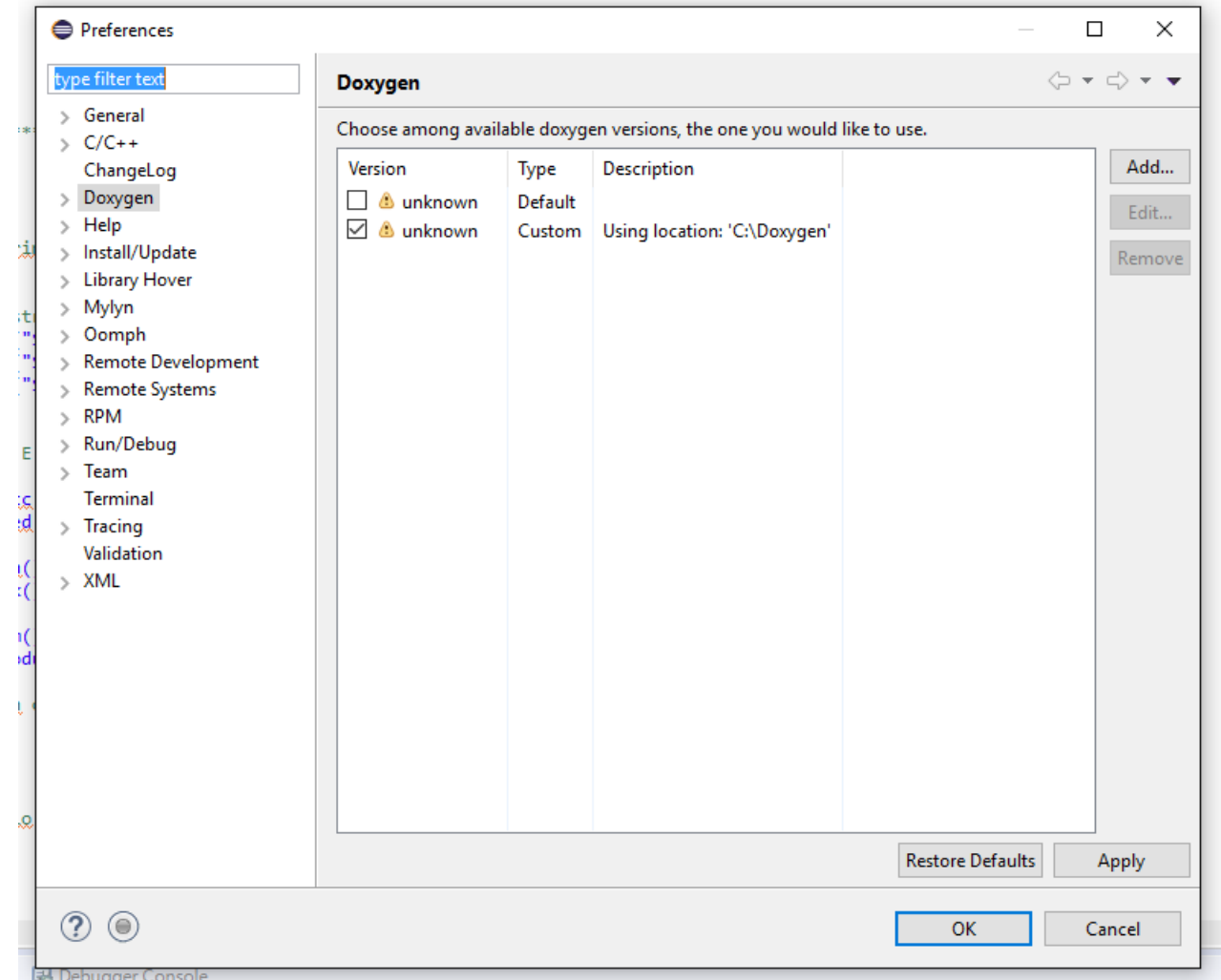
Configurare Doxygen

- Inserire il path di installazione di Doxygen

In Eclipse: **Preferences** → **Doxygen**

Cliccare su **'Add'** per specificare dove si trova l'eseguibile di Doxygen

Aggiungendo il path della cartella che lo contiene **e selezionando quella installazione**

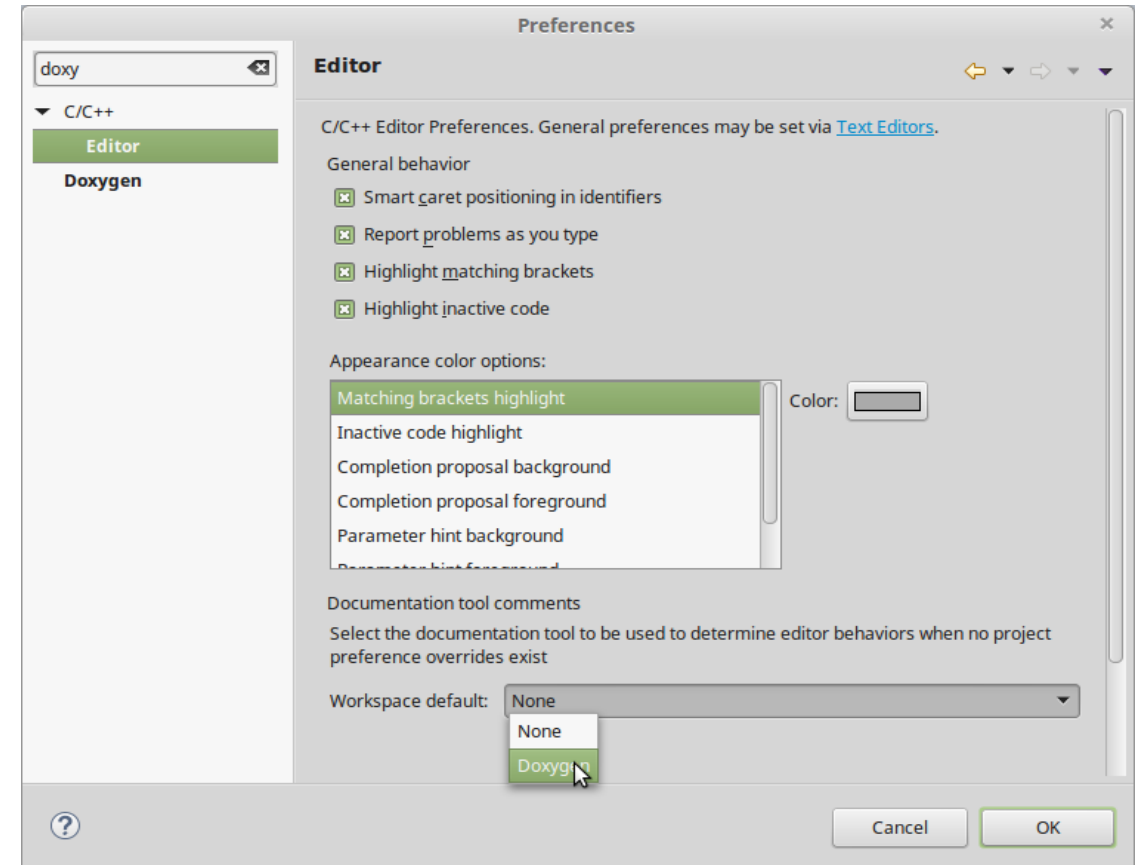


Configurare Doxygen

- Inserire Doxygen nel workspace

In Eclipse: **Preferences** → **C / C++ Editor**

In basso selezionare **'Workspace Default'** e indicare **'Doxygen'** come tool per la generazione della documentazione



Utilizzo di Doxygen in Eclipse

```
/**
 * \file
 * \brief LED driver module.
 * \author Cataldo Musto <musto@uniba.it>
 *
 * This module implements the LED driver, plus a test routine to verify functionality.
 * The module supports right now up to 5 LEDs, and the number of LEDs can be specified using
 * the ::PL_NOF_LED macro in the range of 0 to 5
 * The module has been verified to run both on the SRB and the TOWER platform.
 */

#include "Platform.h" /* interface to the platform */

#if PL_HAS_LED
#include "LED.h"
#include "UTIL1.h"

#define TEST_LED 0
    /*!< if we perform tests */
```

Reference/Hyperlink to symbol

Doxygen comments in blue color

Normal comments in green color

Terminata la configurazione, i **commenti scritti nel formato Doxygen** saranno evidenziati nel **codice sorgente**.

Utilizzo di Doxygen in Eclipse

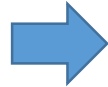
- **Doxygen include anche delle funzionalità per l'autocompletamento dei commenti**
 - Si attiva digitando **“/**”** e premere invio prima di una funzione / procedura
 - Eclox crea il blocco di commento automaticamente, inclusi parametri e valori di ritorno
 - **Crea automaticamente uno «scheletro» di commento da riempire, spiegando a cosa serve ciascun parametro**

```
72 /**  
73  *  
74  * @param currPos  
75  * @param steps  
76  * @return  
77  */  
78 int CalcPos(int currPos, int steps);  
79
```

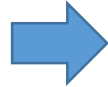
Doxygen: Formattazione commenti

- Commenti “semplici” in C

```
// singola linea
```



```
/*  
 * multi  
 * linea  
 */
```



Commenti **Doxygen** in C

```
/// singola linea
```

```
/**  
 * multi  
 * linea  
 */
```

I commenti in Doxygen hanno una struttura leggermente differente.
Serve a farli distinguere ad Eclipse dai commenti «semplici»

Doxygen: documentare un header file

```
/**  
 * @file documentazione.h  
 * Synthetic description.  
 *  
 * Detailed description here.  
 * @version 0.1  
 * @date 17/mag/2018  
 * @authors c. musto  
 * @copyright GNU Public License.  
 * @bug Not all memory is freed when deleting an object.  
 * @warning Improper use can crash your application  
 */
```

Una caratteristica di **Doxygen** è l'utilizzo di particolari «annotazioni» (tag) che servono a indicare porzioni particolari del codice sorgente.

@file ad esempio indica il nome del file, **@version** la versione. L'uso dei tag serve a dare un «significato» ai vostri commenti

Doxygen: documentare #define e tipi di dato

```
/**  
 * Max admissible value.  
 */  
#define MAX_VALUE 100
```

```
/**  
 * 32-bit unsigned integer.  
 */  
typedef unsigned int UINT32;
```

Nei commenti inseriamo informazioni **sul significato della #define** e sulle **caratteristiche del tipo di dato**

Doxygen: documentare una enumerazione

```
/** An enum type.  
    * Detailed documentation  
    */  
enum EnumType  
{  
    int EVa11,      /**< enum value 1 */  
    int EVa12      /**< enum value 2 */  
};
```

Doxygen: documentare una **variabile globale**

```
/**  
 * Contains the last error code.  
 * @warning Not thread safe!  
 */  
int errno;
```

Nel caso delle variabili globali, si usa il tag **@warning** per fornire eventuali informazioni legate all'utilizzo

Doxygen: documentare una funzione

```
/**
 *   Opens a file descriptor.
 *
 * This function opens a file on the file system ...
 *
 * @param[in] pathname The name of the descriptor.
 * @param[in] flags Opening flags.
 *
 * @return file handler.
 */
int open(const char* pathname, int flags);
```

Utilizziamo **@param** per indicare quali parametri utilizza la funzione e **@return** per spiegare che valore restituisce. Prima di questi tag inseriamo anche una **descrizione generale**

Doxygen: separare i commenti

```
/**
 * Calcola il codice di controllo di un numero ISBN.
 *
 * @param[in] isbn_num Il numero ISBN per il quale
calcolare il codice di
 *
                controllo.
 * @param[in] isbn_length Il formato del codice ISBN (10
o 13).
 * @return il codice di controllo.
 */

char isbn_ctrlcode(char[] isbn_num, int isbn_length);
```

A volte può essere utile distinguere i commenti, inserendo una parte di essi nel **.h** e una parte nel **.c**

Nel .h si inseriscono informazioni su parametri, tipi di ritorno e sullo scopo generale della funzione

Nel .c (dedicato all'implementazione) si dettagliano meglio le pre-post condizioni

Doxygen: separare i commenti

```
/**
 * La funzione calcola il codice di controllo secondo lo standard
ISO
 * (http://it.wikipedia.org/wiki/ISBN#Calcolo_.28ISBN-10.29).
 * La funzione restituisce il codice di controllo per codici ISBN
a    10 o 13 cifre.
 *
 * @pre isbn_num ha lunghezza pari a isbn_length - 1.
 * @pre isbn_num contiene solo cifre decimali
 * @pre isbn_length assume valore 10 o 13
 * @post il carattere restituito e' una cifra decimale oppure 'X'
 */
char isbn_ctrlcode(char[] isbn_num, int isbn_length) {
{
    int i = 0;

    char code [MAX] = {'\0'};

    ...
}
```

A volte può essere utile distinguere i commenti, inserendo una parte di essi nel **.h** e una parte nel **.c**

Nel .h si inseriscono informazioni su parametri, tipi di ritorno e sullo scopo generale della funzione

Nel .c (dedicato all'implementazione) si dettagliano meglio le pre-post condizioni

Manuale Doxygen

Per la lista completa di **comandi Doxygen**, consultate:

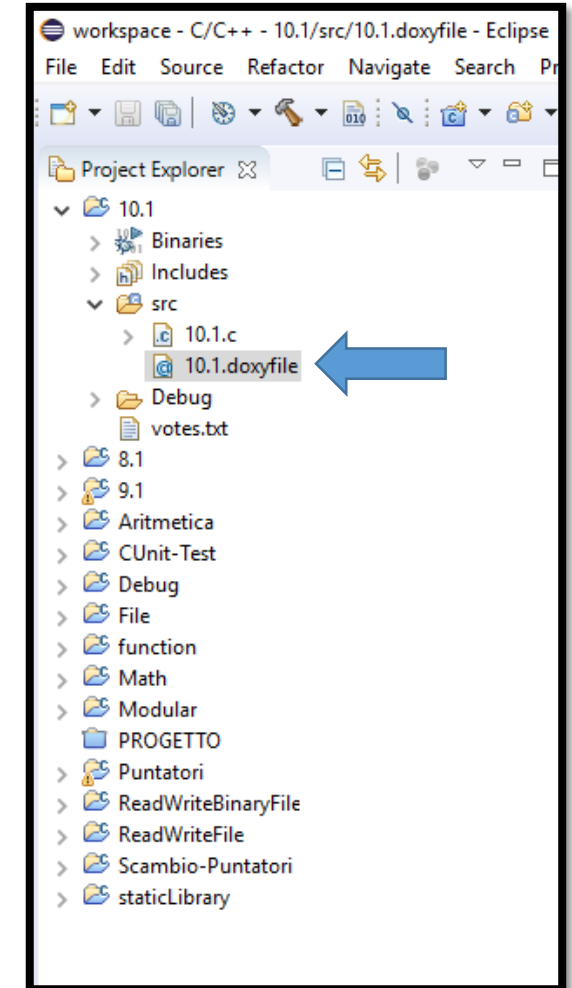
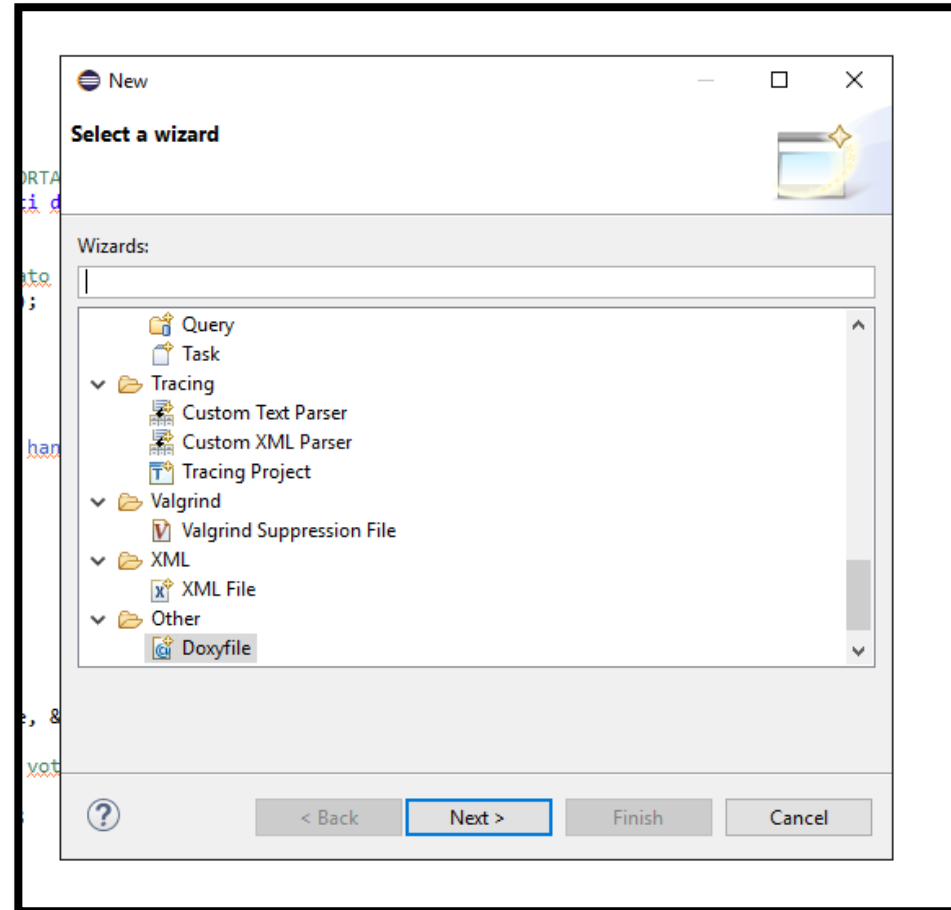
<http://www.stack.nl/~dimitri/doxygen/manual/commands.html>

- **Nota: '@' è equivalente ad '\'**

Analizzate i possibili tag che si possono utilizzare e valutatene l'utilizzo!

Creazione di un Doxyfile

- **Creazione di un doxyfile** in cui inserire tutte le informazioni sul progetto
 - **New → Other → Doxyfile**
- Salvare il file e fare **doppio click**

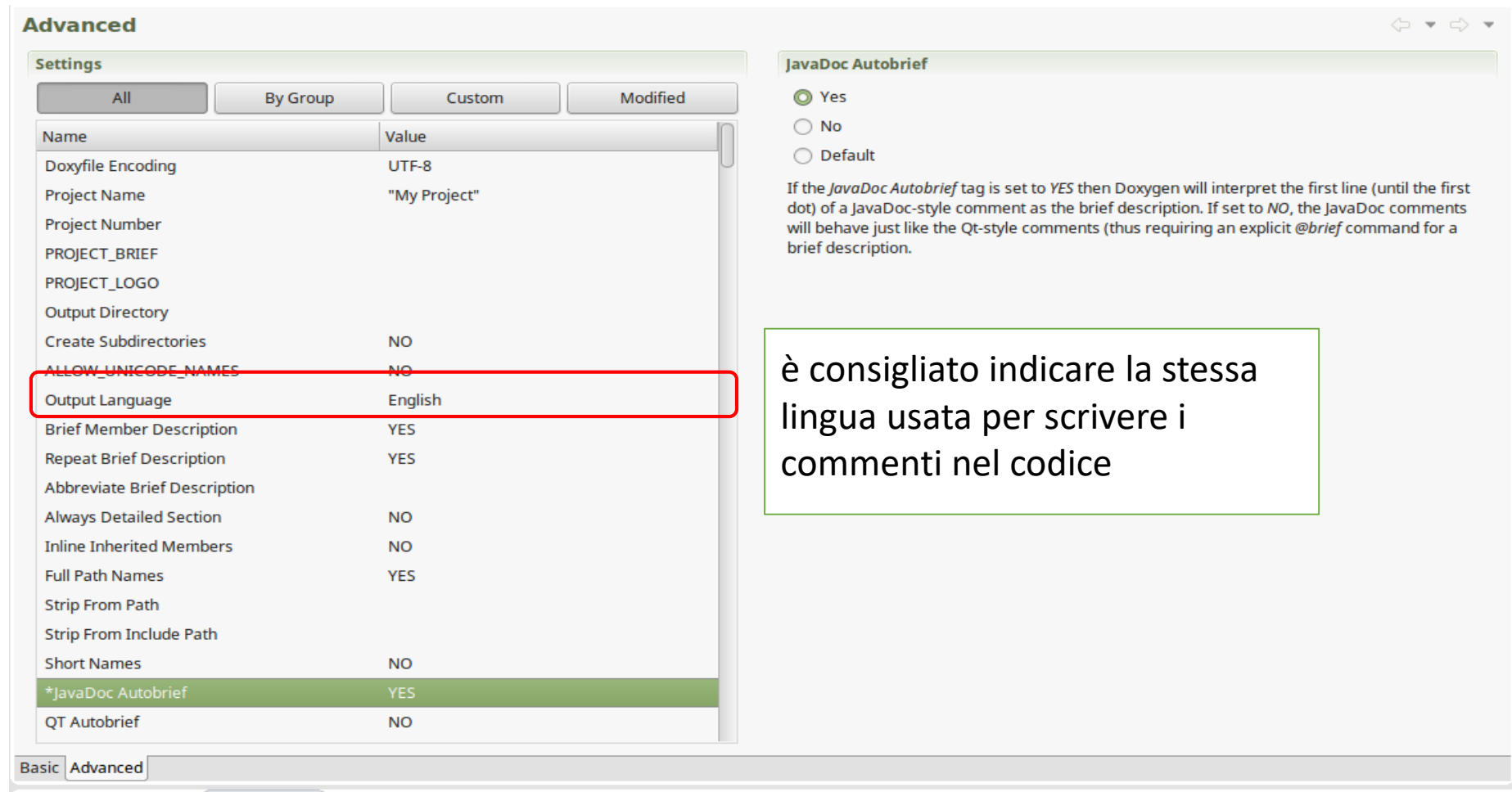


Creazione di un Doxyfile

- **Creazione di un doxyfile** in cui inserire tutte le informazioni sul progetto
 - **New → Other → Doxyfile**
- Inserire i valori per i parametri

The screenshot shows the Doxygen configuration wizard interface. The 'Basic' tab is active. The 'Project' section has 'Name' set to 'My Project' and 'Version or Identifier' set to '1.0'. The 'Input directories' section is empty. The 'Output Formats' section has 'HTML' checked with 'plain HTML' selected, and 'LaTeX' checked with 'as intermediate format for hypedlinked PDF' selected. The 'Diagrams to Generate' section has 'Use built-in diagram generator' selected. The 'Mode' section has 'all entities' selected. The 'Optimize results for:' section has 'C++' selected. The 'Basic' and 'Advanced' tabs are visible at the bottom.

Creazione di un Doxyfile – Configurazione Avanzata



Advanced

Settings

All By Group Custom Modified

Name	Value
Doxyfile Encoding	UTF-8
Project Name	"My Project"
Project Number	
PROJECT_BRIEF	
PROJECT_LOGO	
Output Directory	
Create Subdirectories	NO
ALLOW_UNICODE_NAMES	NO
Output Language	English
Brief Member Description	YES
Repeat Brief Description	YES
Abbreviate Brief Description	
Always Detailed Section	NO
Inline Inherited Members	NO
Full Path Names	YES
Strip From Path	
Strip From Include Path	
Short Names	NO
*JavaDoc Autobrief	YES
QT Autobrief	NO

JavaDoc Autobrief

☒ Yes
☐ No
☐ Default

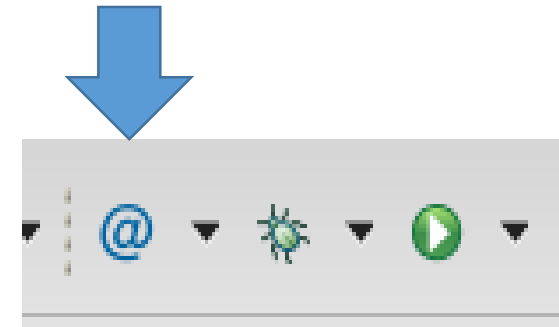
If the *JavaDoc Autobrief* tag is set to *YES* then Doxygen will interpret the first line (until the first dot) of a JavaDoc-style comment as the brief description. If set to *NO*, the JavaDoc comments will behave just like the Qt-style comments (thus requiring an explicit *@brief* command for a brief description).

è consigliato indicare la stessa lingua usata per scrivere i commenti nel codice

Basic Advanced

Generazione della Documentazione

- Cliccare sul pulsante **Doxyfile**
- Nella console parte la **generazione della documentazione**
- Documentazione salvata nella cartella del progetto
- **.../html/index.html**





Esercizio

- **Installare Doxygen + Eclox**
- Selezionare una delle esercitazioni svolte durante il corso, commentarla utilizzando i comandi Doxygen e generare il file
 - A scelta: inviare il file a cataldo.musto@uniba.it
- **Importante:** la documentazione del codice sorgente del caso di studio dovrà essere generata con Doyxgen.
 - Non fare confusione tra **documentazione dell'intero caso di studio** (analisi, progettazione, piano di test, etc.) con la **documentazione del codice sorgente!**