



# SHARIF AI CHALLENGE

---

## 2020

محتوای علمی  
دوازدهمین نبرد هوش مصنوعی شریف

# Debug

زمستان ۱۳۹۸

دیباگ چیه؟!

خب بذارید داستان ورود کلمه bug در دنیای کامپیوتر رو تعریف کنم! (شاید خیلیا هم شنیده باشید!)

در گذشته کامپیوترها خیلی بزرگ بودن. روایت داریم اندازه یه اتاق! بعد اینطوری بود که گاهی دچار مشکل میشد دستگاه. میرفتن بررسی میکردن میدیدن یه مگس = bug وارد اتاق شده و سیستم رو به هم ریخته! لذا از اون تایم به اکثر ایراداتی که در کدها و کامپیوتر رخ میدید میگویند باگ و به کارهایی که جهت رفع اون انجام میشه میگویند debug (تا اینجا خواستم صفحه پر کنم (:

Debugging به فرایند مشکل یابی یک نرم افزار یا اپلیکیشن گفته می شود. زمانی که ما کدنویسی می کنیم، معمولاً در حین کدنویسی برخی خطاها را مرتکب می شویم که در نهایت منجر به این می شوند تا برنامه ی ما آن طور که باید و شاید کار نکند. به طور کلی، ما معمولاً چند نوع مشکل در برنامه های خود داریم که عبارتند از:

**ارورهای سینتکسی:** به نوشتار کدهای یک زبان برنامه نویسی Syntax گفته می شود. گاهی اوقات برنامه نویسان در حین

نوشتن برخی دستورات، غلط املائی مرتکب می شوند. مثلاً به جای نوشتن دستور print، می نویسند pritrn. در برخی زبان ها

مثل HTML و CSS مرتکب شدن چنین خطاهایی خیلی مشکل ساز نیست اما برخی از دیگر زبان ها مثل PHP و Python با ارورهای سنتکسی خیلی مشکل دارند و برنامه هرگز اجرا نخواهد شد.

**ارورهای سمنتیک:** این دست ارورها زمانی رخ می دهند که کدهای شما درست است اما نتیجه ی مطلوب را دریافت نمی کنید

مثل زمانی که عددی را بر صفر تقسیم کنید (در ریاضیات چنین عملی امکان پذیر نیست!)

**ارورهای منطقی:** این دست از ارورها یا مشکلات جزو ارورهای سخت هستند و شاید یک برنامه نویس روزها و شاید هفته ها برای

یافتن آنها می بایست زمان بگذارد. سینتکس برنامه درست است و برنامه می بایست همان طور که انتظار می رود اجرا شود اما

# AI CHALLENGE

## 2020

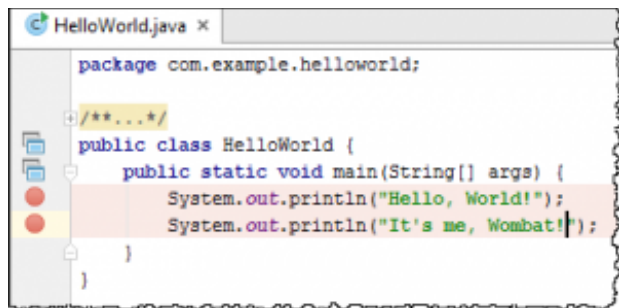
واقعیت این گونه نیست! فرض کنیم که یک فروشگاه آنلاین داریم و زمانی که مشتری به سبد خرید خود می رود،

برنامه‌ای که ما نوشته‌ایم جمع سبد خرید را اشتباه در معرض دید وی قرار می دهد.

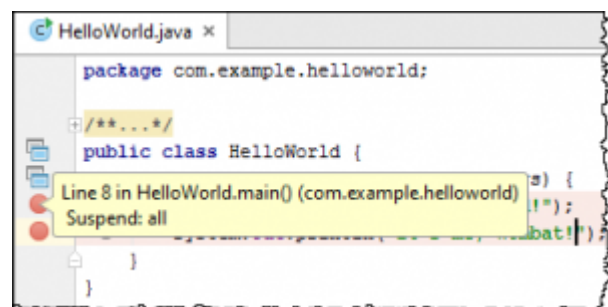
### *JetBrains ( IntelliJ(Java) / PyCharm(Python) / CLion(C) )*

قرار دادن نقطه توقف (breakpoint): برای شروع دیباگ ابتدا باید یک breakpoint در دستوراتی که شما میخواهید در آن ها

اجرای برنامه خود را متوقف کنید ، قرار دهید. در مثال زیر و کد موجود ما breakpoint را در ابتدای دو دستور قرار میدهم.



رنگ خط مشخص شده تغییر میکند. اگر ماوس را روی نقطه قرمز قرار دهید، خصوصیت آن در tooltip دیده میشود.



اگر میخواهید ویژگی های این breakpoint را تغییر دهید روی آن کلیک راست میکنید

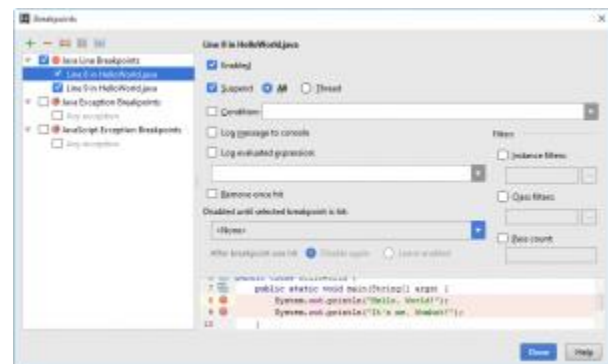


# AI CHALLENGE

## 2020

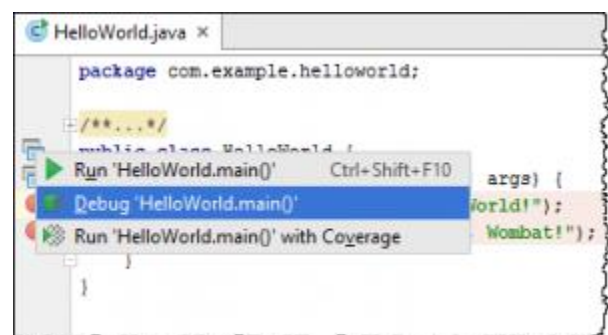


برای دیدن همه نقاط توقف و تغییر ویژگی های آنها کافیست  $\text{Ctrl} + \text{Shift} + \text{F8}$  را فشار دهید



شروع دیباگ

با کلیک روی breakpoint منویی باز میشود که از طریق آن میتوانید کد را run یا debug کنید

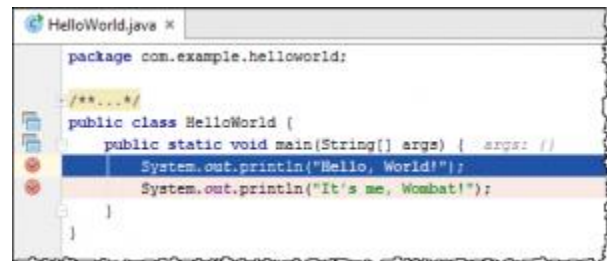


# AI CHALLENGE

## 2020

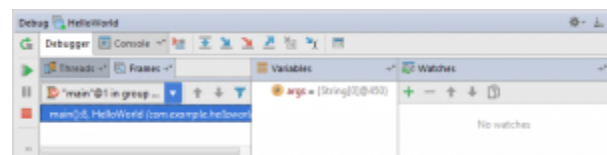
کد شما کامپایل میشود (ممکن است طول بکشد!!) و سپس برنامه در اولین breakpoint متوقف میشود. تغییراتی

مشاهده میشود از جمله تغییر رنگ خط breakpoint



به این معنی است که برنامه تا قبل این خط اجرا شده و در این دستور معلق شده است. در قسمت پایین صفحه IntelliJ یک

پنجره به نام debug tool window ظاهر میشود که تمام ابزارهای مورد نیاز برای انجام دیباگ در آن قابل مشاهده است.



حال دیباگ را ادامه میدهیم. کافیست روی فلش آبی رو به پایین نوار ابزار در پنجره debug tool window کلیک کنید یا  $F^8$

را فشار دهید. اکنون خط بعدی آبی شده و تغییرات زیر اعمال میشود.

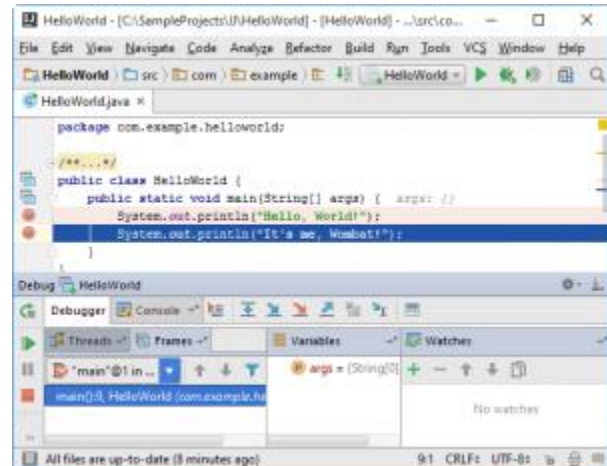
۱- در کادر frames شماره خط بعدی نمایش داده میشود.

۲- در کادر console یک آیکون زرد رنگ اضافه شده که نشان میدهد برنامه دارای یک خروجی جدید است.

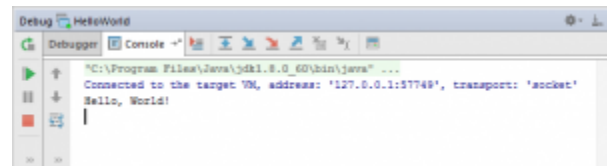


# AI CHALLENGE

## 2020



روی زبانه!!(tab) console کلیک کنید تا خروجی را ببینید



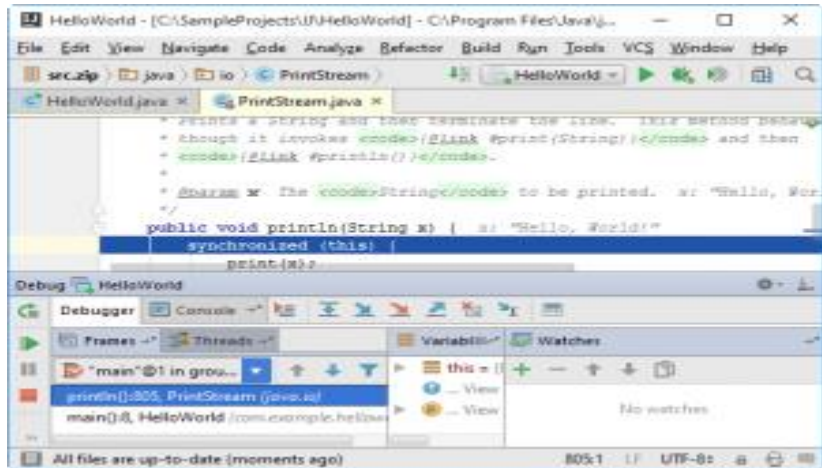
با فشردن  $F^A$  یا کلیک روی فلش آبی رو به پایین دستور بعدی هم اجرا شده و با تکرار این کار برنامه پایان میابد.

حال روشی با اندکی پیچیدگی بیشتر معرفی میکنیم! کافیست پس از شروع دیباگ و رسیدن به breakpoint دکمه های

$Shift+Alt+F^V$  را فشار دهید تا به درون تابع `println` در کلاس `PrintStream` برویم.

# AI CHALLENGE

## 2020



حال با فشردن  $\text{Shift} + \text{F}^8$  به نقطه توقف بعدی میرویم.

خب این یک جلسه خیلیییی ساده از دیباگ بود! قطعاً کدهای شما اینقدر آسان نیست :

کارهای مختلفی که میتوان انجام داد :

**متوقف کردن دیباگ:** از طریق منوی اصلی گزینه های **Run | Pause Program** را انتخاب کنید. / در نوار ابزار دیباگ،

|| را کلیک کنید.

**ادامه دادن دیباگ:** از طریق منوی اصلی گزینه های **Run | Resume Program** را انتخاب کنید. / در نوار ابزار  را

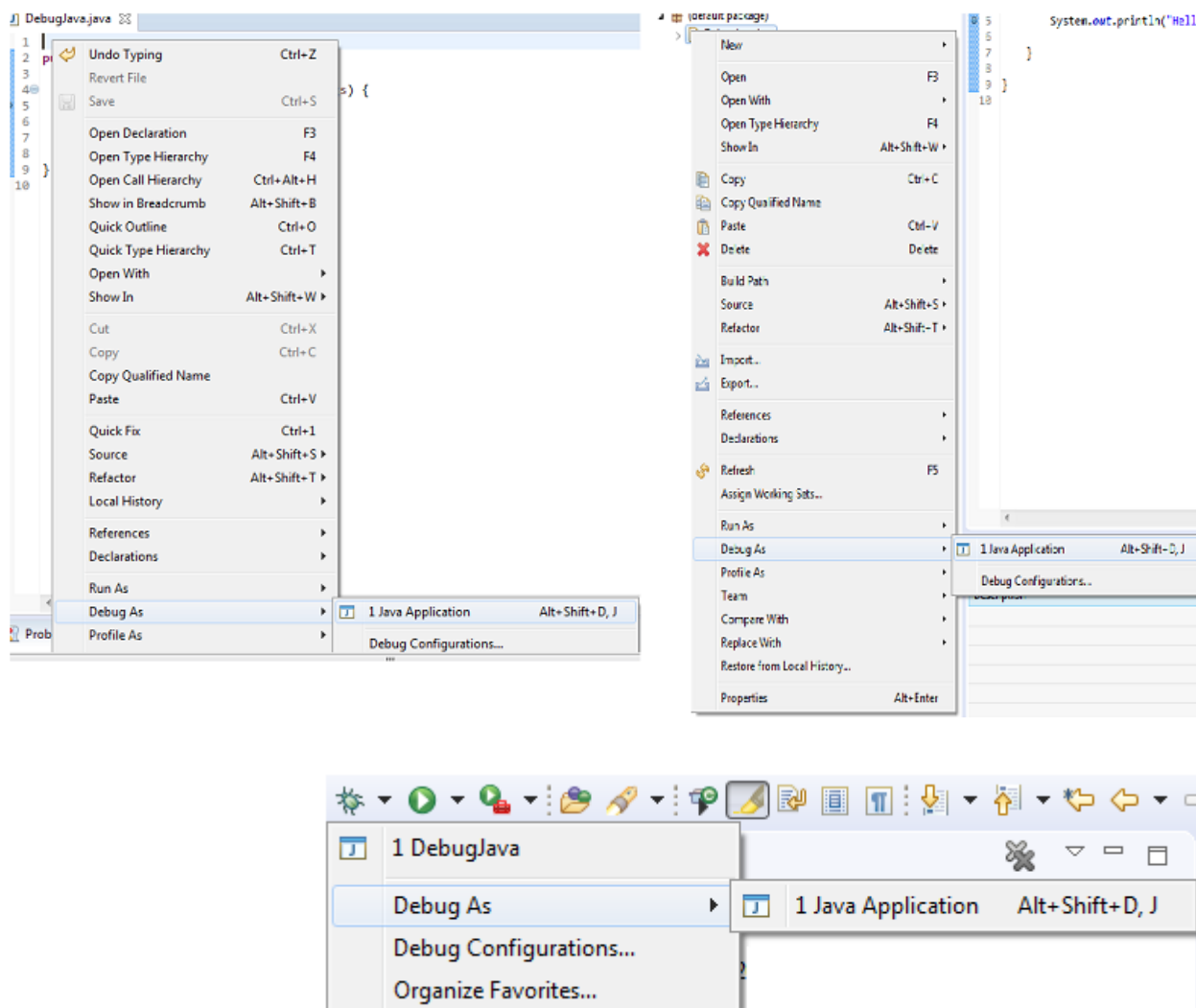
کلیک کنید. / دکمه  $\text{F}^9$  را فشار دهید.

## Eclipse

برای شروع دیباگ کافیت روی نام کلاس مدنظر در قسمت Package explorer راست کلیک کنید و گزینه

# AI CHALLENGE 2020

Alt + Shift + D, J یا Debug As → Java Application را فشار دهید.

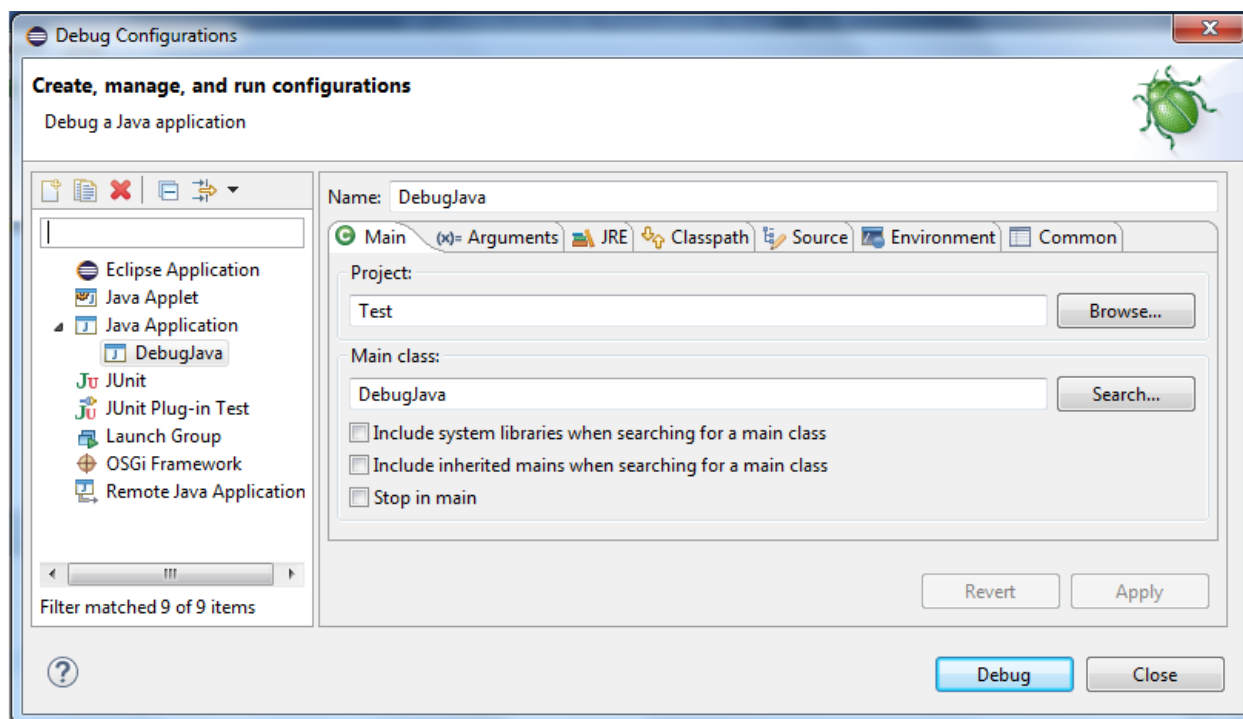


هریک از کارهای فوق یک **Debug Launch Configuration** جدید ساخته و با استفاده از آن میتوان دیباگ را شروع کرد.



# AI CHALLENGE

## 2020

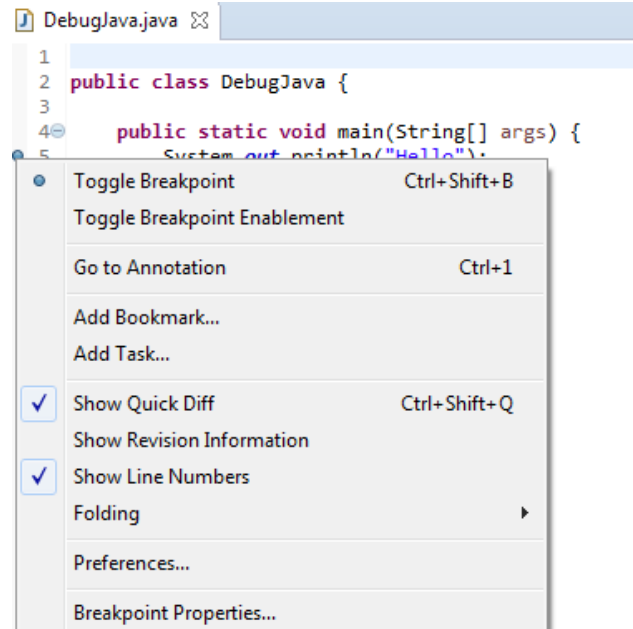


حال باید breakpoint بسازیم. (مانند jetbrains ها ! ) با دوبار کلیک کردن بر خط موردنظر یا کلیک راست و انتخاب

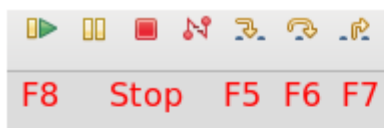
Toggle Breakpoint میتوان روی خط مورد نظر breakpoint ساخت.

# AI CHALLENGE

## 2020



ابزارهای زیر در eclipse برای دیباگ کردن تعبیه شده اند.



Shortcut	Toolbar	Description
F5 (Step Into)		Steps into the call
F6 (Step Over)		Steps over the call
F7 (Step Return)		Steps out to the caller
F8 (Resume)		Resumes the execution
Ctrl + R (Run to Line)		Run to the line number of the current caret position
Drop to Frame		Rerun a part of your program
Shift + F5 ( Use Step Filters)		Skipping the packages for Step into
Ctrl + F5 / Ctrl + Alt + Click		Step Into Selection

برای دیدن ابزارهای بیشتر و پیشرفته تر از لینک زیر و مطالب انتهایی آن استفاده کنید. (:



# AI CHALLENGE

---

## 2020

[https://www.eclipse.org/community/eclipse\\_newsletter/۲۰۱۷/june/article۱.php](https://www.eclipse.org/community/eclipse_newsletter/۲۰۱۷/june/article۱.php)

### *NetBeans*

بعید میدونم تعداد افرادی که از این ide استفاده میکنن زیاد باشه! در کل ide خوبی هست و امکانات گرافیکی خوبی داره. صرفاً

لینک یوتیوب زیر رو معرفی میکنیم که خیلی کوتاه دیباگ در NetBeans رو بررسی کرده. (باشد که رستگار شوید!)

<https://www.youtube.com/watch?v=۲Z۹B۰wYhKWw>