

Lenguajes de Programación, 2023-1

Instalación de RACKET

Profesora: Karla Ramírez Pulido
Ayud. Lab.: Manuel Ignacio Castillo López
Facultad de Ciencias, UNAM

Introducción

En estas notas, se presentan varias formas de instalar e interactuar con RACKET para comenzar el desarrollo en esta plataforma. El alcance de este documento se limita a la instalación e interacción básica con los entornos de desarrollo e intérpretes del lenguaje; por lo que se deja al lector estudiar por su cuenta las varias herramientas con las que dispone la plataforma, como los distintos lenguajes y bibliotecas con los que podemos trabajar. «The Racket Guide» (vea las referencias) y la documentación en línea (<https://docs.racket-lang.org/>) ofrecen la mayoría de la información necesaria para aprender a trabajar con RACKET.

Se presentan dos interfaces para trabajar con RACKET: DrRacket y el intérprete de línea de comandos. DrRacket es un IDE (*Integrated Development Environment* - Entorno de Desarrollo Integrado) que ofrece un intérprete REPL (*Read-Eval-Print Loop* - Repeticiones de Lectura-Evaluación-Impresión); es decir, podemos ingresar instrucciones y dinámicamente obtener resultados. Contiene un área de trabajo donde podemos ingresar programas más completos que una sola instrucción. DrRacket también nos ofrece herramientas como cambiar el lenguaje o editar múltiples programas como pestañas del área de trabajo.

Por su parte, el intérprete en línea de comandos ofrece un modo interactivo o puede solo ejecutar una instrucción o programa y salir. El modo interactivo es muy similar al REPL como aparece en DrRacket. En ambos intérpretes, podemos cargar programas que hemos creado anteriormente; como veremos al final de estas notas. Nos referiremos al intérprete de línea de comandos como «Intérprete ILC» pues a esta clase de programas se les suele llamar «CLI» (*Comand Line Interface*); o en español, Interfaz de Línea de Comandos (ILC).

Instalación

RACKET está disponible las plataformas Microsoft Windows, Apple Mac OS X y otros sistemas POSIX-compatibles; principalmente distribuciones de Linux. Las distintas variantes de RACKET están disponibles en: <https://download.racket-lang.org/>

En esta sección se muestra cómo instalar RACKET en Windows y Linux; para instalar en Mac OS X, debe seguir un procedimiento muy similar al de Windows, eligiendo la variante adecuada para su sistema (Intel 32-bits, Intel 64-bits o Apple Silicon para arquitecturas M1, M2 y otras compatibles). Únicamente se deben seguir los pasos del instalador. Sin embargo, el instalador no agrega una referencia al archivo en código binario de RACKET a las variables de entorno (PATH), por lo que si desea invocar el intérprete de RACKET o el gestor de paquetes que incluye el instalador (raco) desde una ubicación que no es la de instalación, deberá seguir las siguientes instrucciones: <https://beautifulracket.com/setting-the-mac-os-path.html>

Tome en cuenta que la ubicación recomendada para instalar Racket es «/Applications/Racket vX.X»; ponga atención en la ruta que indique el instalador o a use la misma ruta personalizada que haya indicado en el instalador.

Microsoft Windows

Comenzaremos por ingresar a la página de descarga indicada en el primer párrafo de la sección anterior. Elegiremos nuestra plataforma Windows x64/x86 dependiendo de la arquitectura del *Sistema Operativo* (si su CPU es de 64 bits, pero Windows es de 32, no podrá ejecutar aplicaciones de 64 bits). Por ejemplo, en la Figura 1 se muestra la página de descarga en la que se ha elegido el instalador para Windows x64.

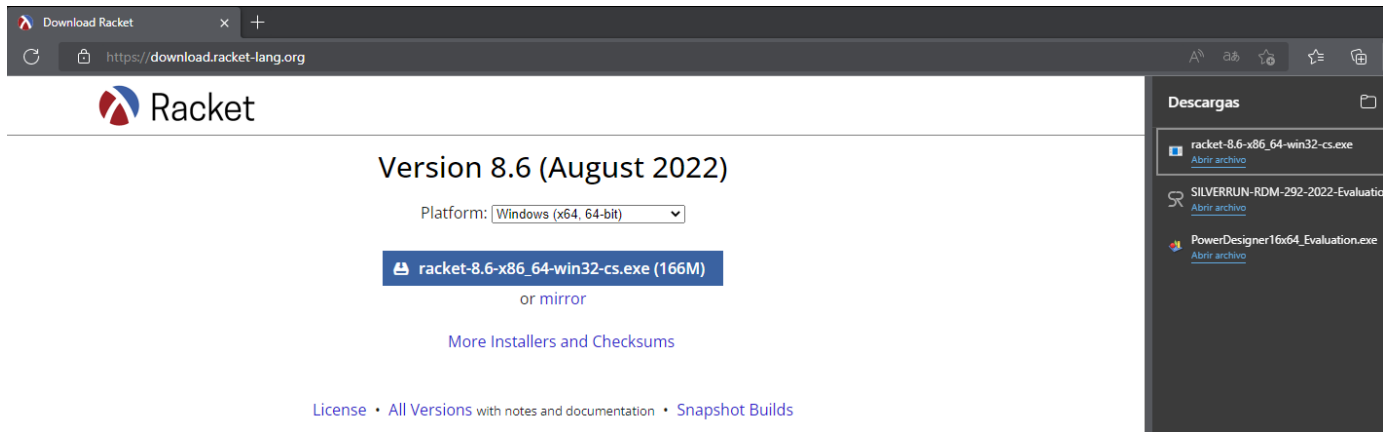


Figura 1: Pantalla de descarga de RACKET. Elija la versión adecuada para su sistema.

El siguiente paso es ejecutar el instalador que acabamos de descargar y seguir los pasos que nos muestra. Es importante recordar la ruta de instalación que solicitamos o que aparece por defecto; pues al final tendremos que agregar la ruta a RACKET en las variables de entorno (PATH). La ruta de instalación se elige en la pantalla que aparece en la Figura 2.

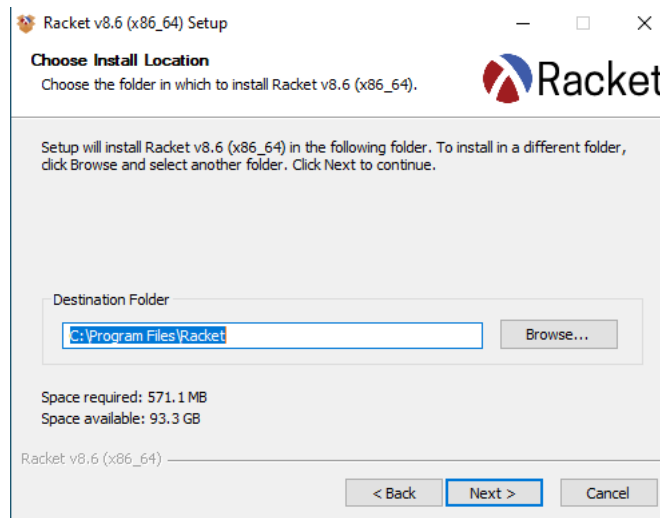
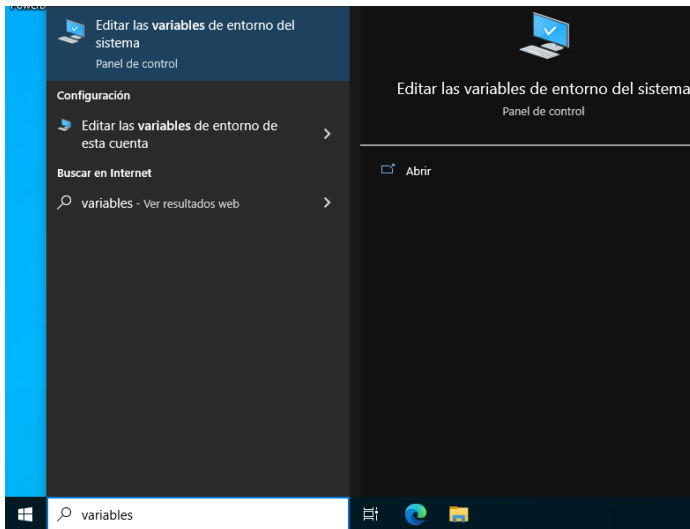


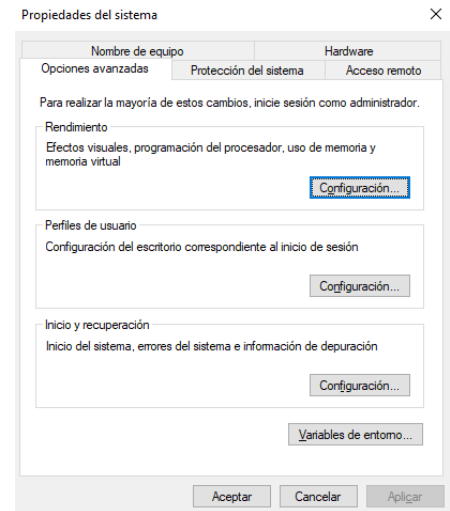
Figura 2: Instalador de RACKET para Windows. Observe y anote la ruta de instalación elegida.

Una vez que el instalador ha finalizado, lo cerramos y procedemos a agregar RACKET al PATH. Para esto, podemos usar la barra de búsqueda en el menú inicio e ingresamos el texto de búsqueda «Editar las variables de entorno del sistema»; tal como se muestra en la Figura 3a. Esto abrirá el cuadro de diálogo «Propiedades del sistema» en la pestaña «Opciones avanzadas» que aparece en la Figura 3b. En esta pestaña seleccionaremos el botón «Variables de entorno».

En cuadro de la Figura 4a; en el apartado de variables de entorno del usuario, ubicaremos la variable PATH que debemos editar. Aparecerá otro cuadro emergente donde se listan las rutas que forman parte de PATH, agregaremos



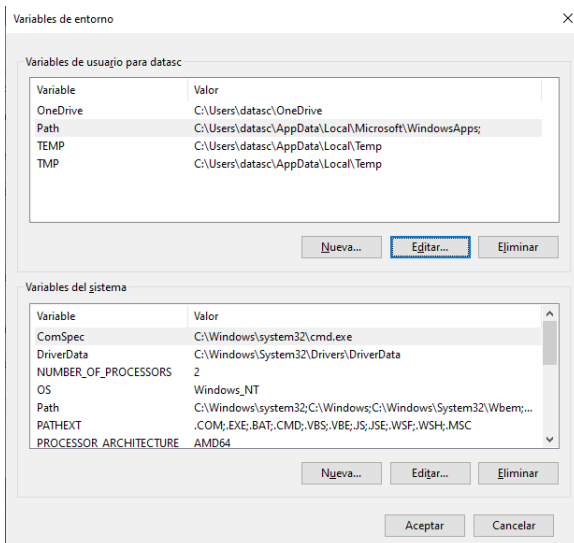
(a) Use la herramienta de búsqueda en la barra de tareas para abrir las propiedades del sistema



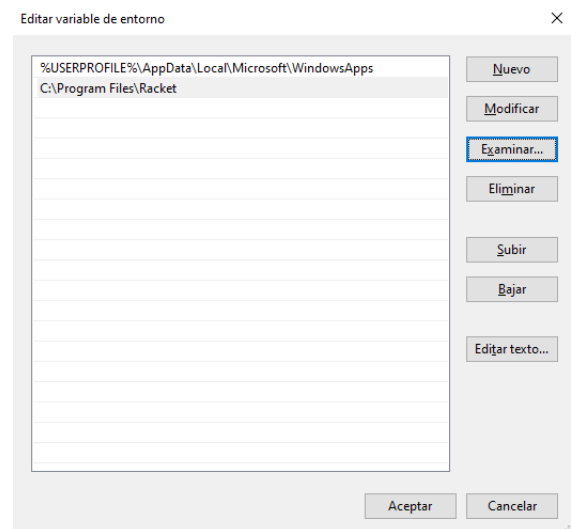
(b) Cuadro de diálogo «Propiedades del sistema» en la pestaña «Opciones avanzadas»

Figura 3: Abriendo el editor de Variables de Entorno

una nueva. Podemos seleccionar el botón «Nuevo» para ingresar manualmente la ruta que indicamos en nuestro instalador, o usar el botón «Examinar» para localizar el directorio donde se encuentra el código *binario* (no la carpeta de instalación, la carpeta donde está el binario) de RACKET. En la versión 8.6 para Windows x64, el binario se encuentre en la misma carpeta de instalación. Al agregar la ruta, el cuadro de diálogo de edición de PATH debe verse cómo en la Figura 4b.



(a) Eliga la variable de usuario PATH



(b) Agregando RACKET a la variable PATH

Figura 4: Editor de variables de entorno

Ahora podrá iniciar la línea de comandos CMD o POWERSHELL y debería poder usar RACKET desde cualquier ubicación, como se muestra en la Figura 5.

DrRacket también estará disponible, y una vez que el archivo en código binario está disponible desde PATH, no debería tener ningún problema para empezar a crear programas en RACKET.

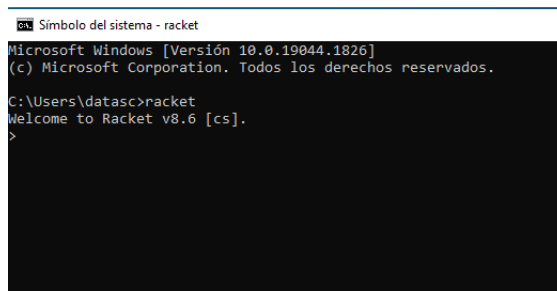


Figura 5: Resultado de probar ejecutar Racket desde la ubicación por defecto para el usuario local

Distribuciones de Linux y otros sistemas POSIX-compatibles

Antes de descargar RACKET de la dirección indicada en la sección anterior, compruebe que existan repositorios de RACKET para su distribución; de manera que pueda realizar la instalación de forma simple a través de su gestor de paquetes. Por ejemplo, existen repositorios de RACKET para Ubuntu y Debian, en el caso particular de Debian son parte de los repositorios principales, por lo que no es necesario realizar ninguna configuración adicional. Por ejemplo, en distribuciones basadas en Debian (como Ubuntu, Kali, Knoppix...), puede realizar la instalación con apt de la siguiente manera:

```
sudo apt install racket
```

En caso de que su distribución no cuente con repositorios de RACKET, puede descargar el instalador `.sh` que deberá ejecutar desde consola; si desea hacerlo disponible a todos los usuarios posiblemente necesite ejecutar el instalador como súper-usuario. Si usa un sistema POSIX-compatible no basado en Linux (o Mac OS X); puede descargar el `.tgz` con el código fuente para realizar su propia compilación. Sólo en estos dos casos se deberá también agregar el directorio con el binario de RACKET a sus variables de entorno; si realiza la instalación con un gestor de paquetes a través de un repositorio, esto ocurre de forma automática.

Creando programas en Racket

En esta sección, veremos brevemente cómo crear un programa en RACKET; ya sea que desee trabajar con DrRacket o con el intérprete ILC. Supondremos que ha realizado el proceso de instalación para su plataforma conforme a la sección anterior.

DrRacket

Lo primero que debemos hacer al iniciar DrRacket, es elegir un lenguaje. Estos básicamente son dialectos de Racket; que podemos interpretar como *marcos de trabajo* o bibliotecas, la diferencia principal con el lenguaje racket es que importan por defecto una serie de módulos que dependen del lenguaje elegido. En este curso, vamos a utilizar el lenguaje `plai` que ofrece una serie de extensiones que aprovecharemos para desarrollar la parte práctica del curso. Para establecer el lenguaje `plai` en DrRacket comenzamos por seleccionar en el la barra de menú

Language > Choose Language ...

Se muestra un cuadro de diálogo (cómo el que aparece en la Figura 6) que nos pide elegir la categoría de lenguaje con la que vamos a trabajar. Seleccionaremos la opción «*Racket Language*»

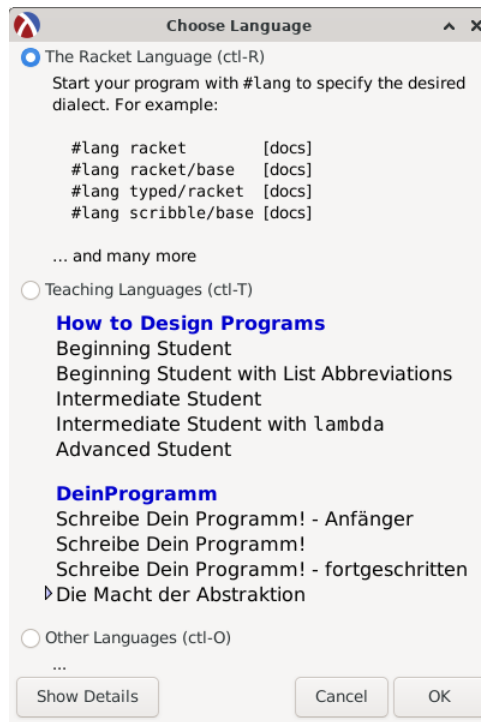


Figura 6: Cuadro de diálogo para elegir la categoría de lenguajes con las que podemos configurar DrRacket

Una vez que elegimos la opción indicada en la Figura 6, DrRacket sabrá que trabajaremos con variantes de racket y al ejecutar nuestros programas, usará el intérprete de alguno de sus dialectos. Sin embargo, aún desconoce específicamente cuál será. En general; siempre que creemos un programa con RACKET, es importante indicar el dialecto en la primera línea. Puesto que vamos a trabajar con plai, en el área de texto de DrRacket vamos a poner en la primera línea que queremos trabajar con plai; tal como se muestra en la Figura 7.

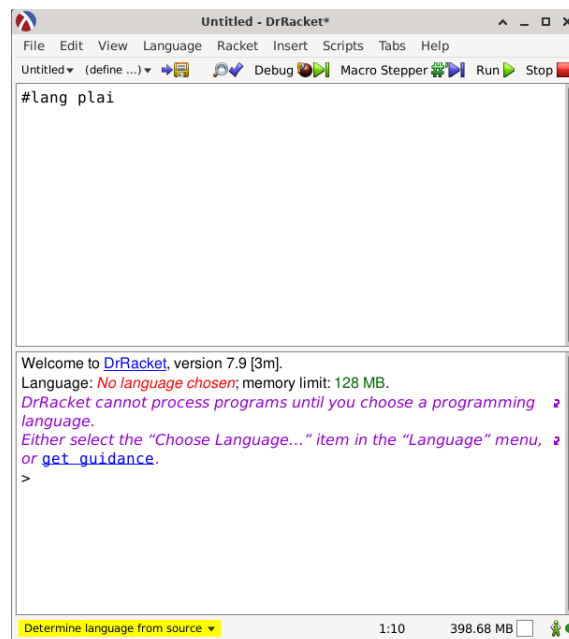


Figura 7: DrRacket configurado para utilizar el lenguaje plai

Con esto estamos listos para empezar a crear programas con DrRacket con el lenguaje `plai`. La Figura 8 muestra un ejemplo de una definición una vez que se ha completado la configuración expuesta en esta sección.

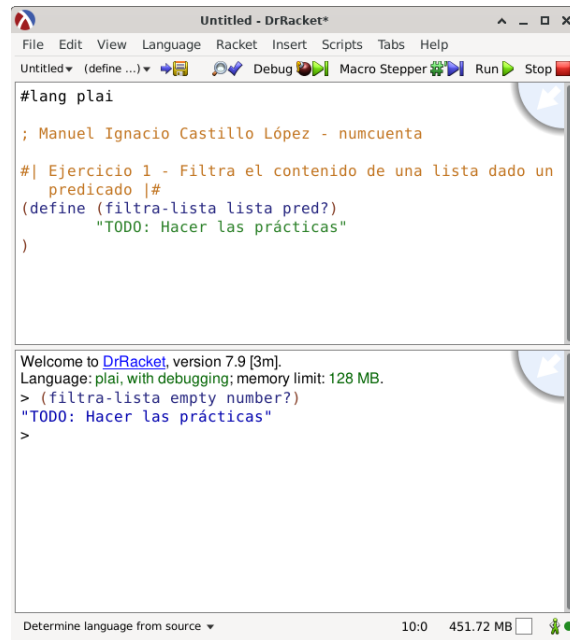


Figura 8: Ejemplo de uso de DrRacket con la configuración completa

Para ejecutar nuestros programas, únicamente necesitamos seleccionar el botón `run`; o bien, podemos usar el atajo de teclado `ctrl + R`.

Intérprete ILC

Cuando trabajamos con el intérprete ILC de RACKET no es necesario configurar de antemano el lenguaje o familia de lenguajes con las que vamos a trabajar. En este caso, solo es necesario iniciar en la primer línea de nuestros programas con la instrucción `#lang plai`, por ejemplo:

```
1 #lang plai
2
3 ; Manuel Ignacio Castillo López — numcuenta
4
5 #| Ejercicio 1 — Filtra el contenido de una lista dado un predicado |#
6 (define (filtra-lista lista pred?)
7   "TODO: Hacer las prácticas"
8 )
```

Para ejecutar nuestros programas, podemos indicar el nombre (con ruta si es necesario) del archivo con la bandera `-t`, la cuál toma una cadena que interpreta como ruta de archivo, lo importa como módulo y lo ejecuta.

```
racket -t mi-programa.rkt
```

Sin embargo, esto solo ejecuta la interpretación del script y termina; no podemos interactuar con las definiciones en el programa. Podemos cargar el programa como un módulo (paquete, dependencia) en el modo interactivo del intérprete usando la instrucción `(require "ruta/a/mi-programa.rkt")`; por ejemplo:

```
user@host:~$ racket
Welcome to Racket v7.9 [bc].
> (require "mi-programa.rkt")
>
```

Si al seleccionar la tecla «enter» no obtenemos errores, quiere decir que el programa ha sido encontrado en el sistema de archivos exitosamente y que su interpretación no arrojó errores; por lo que estamos listos para explotar sus definiciones.

En sistemas POSIX-compatibles, también podemos hacer nuestros programas scripts auto-ejecutables. Para esto, necesitamos dar permisos de ejecución (`chmod`) a nuestro archivo con nuestro programa y como primer línea; inmediatamente **antes** de `#lang plai`, ponemos un shebang con una ruta al entorno donde exista un enlace al binario del intérprete de RACKET. En sistemas basados en Debian, por ejemplo, podemos usar

```
#!/usr/bin/env racket
```

De esta manera podemos ejecutar el script sin tener que invocarlo como un argumento del intérprete o importándolo como módulo; pero no afecta estas cargas como módulo, por lo que aún podemos cargarlo como dependencia.

Guardando y recuperando programas

Cuando creamos un programa en RACKET; como sugiere la sección anterior, no estamos obligados a escribir nuestro programa cada vez que queramos usarlo o modificarlo. Dependiendo cómo estemos usando RACKET, seguiremos pasos diferentes para lograrlo; como los que se describen a continuación. En general, los archivos que representan scripts de RACKET usan la extensión `.rkt`

DrRacket

Una vez que hemos escrito contenido en el área de trabajo de DrRacket, podemos guardarlo como un archivo de texto (con extensión `.rkt`) para continuar modificándolo más adelante o utilizar sus definiciones en el REPL. Para esto, bastará con elegir la opción en la barra de herramientas

```
File > Save Definitions As...
```

También podemos usar el atajo de teclado `ctrl + shift + S`; DrRacket nos mostrará un cuadro de diálogo con el que podremos elegir la ubicación y nombre de nuestro archivo (recuerde que por convención la extensión es `.rkt`). DrRacket guardará nuestro programa en el archivo indicado y podremos continuar haciendo definiciones. En adelante, para sobrescribir los cambios que haga en el área de definiciones de DrRacket, puede usar la opción en la barra de herramientas

```
File > Save Definitions
```

O el atajo de teclado `ctrl + S` para no tener que navegar por el sistema de archivos cada vez que que desee guardar sus cambios. Como con muchos otros editores, se le recomienda guardar sus cambios de forma periódica durante sus sesiones de trabajo.

Posteriormente; en una nueva sesión con DrRacket o tras cerrar la pestaña en la que estábamos editando nuestro programa, podemos recuperarlo con la opción en la barra de herramientas

File > Open...

O con el atajo de teclado `ctrl + O`; que abrirá un cuadro de diálogo similar al anterior con el que podemos elegir el archivo `.rkt` que contiene nuestro programa. Al elegirlo, se cargará su contenido en el área de trabajo y podemos continuar editándolo o interactuando con él en el REPL al ejecutarlo con `ctrl + R` (Run).

Intérprete ILC

Si se utiliza el intérprete ILC deberá escribir sus programas utilizando un editor de texto para poder guardarlos; ya que el intérprete no incluye mecanismos que permitan volcar las definiciones (vaciar la memoria del intérprete) hechas durante la sesión en archivo de forma directa. La estrategia de trabajo que recomendamos es crear sus programas en un archivo de texto con extensión `.rkt` e importarlo como módulo en el intérprete ILC (recuerde que para esto usamos (`require "ruta/a/mi-programa.rkt"`)) para verificar su integridad sintáctica y el comportamiento de las definiciones (es posible que tenga que cerrar el intérprete con (`exit`) para evitar conflictos por definiciones repetidas). Básicamente se trata de la misma estrategia que al trabajar con `DrRacket`, solo que en lugar de usar la función de ejecución (Run), cargaremos el archivo en el intérprete ILC.

Por lo anterior, al editar su programa con un editor de texto debería guardar todos los cambios realizados en su archivo `.rkt` para poder correr su última versión en el intérprete.

También podría integrar el intérprete de `RACKET` con un editor de texto especializado; como `atom`, o con el IDE de su preferencia.

Referencias

- [1] M. Flatt y R. B. Findler, «The Racket Guide», PLT Research Group, 8.5.0.8, jul. 2022.
- [2] Racket, «1 Installation», PLT Research Group, en línea <https://docs.racket-lang.org/pollen/Installation.html>
Consultado el 10 de agosto de 2022.