

TECNOLÓGICO DE COSTA RICA
Escuela de Ingeniería en Computación
Proyecto de Ingeniería de Software

Profesora:
Maria Estrada

Entrega 1:
Resumen reglas de estilos

Estudiantes:
Christian Leon Guevara - 2013371982
Gabriel Ramírez Ramírez - 201020244

Fecha de entrega:
22-12-2018

Periodo Verano

Tabla de contenido

Introducción.	3
Lista de reglas de estilo.	3
• Nombres: La regla principal es el uso de nombres significativos para hacer el código entendible.	3
• Comentarios.	3
• Espaciado: Es necesario siempre separar las palabras con espacios en blanco, nunca debe hacerse usando tabulaciones.	3
• Formato: A continuación, se presentan algunos casos particulares de cómo se debe escribir el código.	4
• Identación: La indentación es importante ya que permite una mejor legibilidad del código.	4

Introducción.

En este documento se establecen las principales reglas de estilos que se deben seguir a la hora de desarrollar código en OCAML. Son reglas establecidas por distintas fuentes, sin embargo, en este documento se establecen las que llegan a un consenso entre todas.

Lista de reglas de estilo.

Las reglas de estilos es una guía de cómo se debe escribir código, en nuestro caso particular el lenguaje OCAML. Dichas reglas pueden definir convenciones para: escribir nombres, comentarios, espaciado, formato e indentación.

El principal principio a la hora de escribir los programas es asegurarse de que sea sencillo y fácil de leer. Ya que la mayor la cantidad de tiempo se dedica a la lectura de código y hacerlo cumpliendo el principio permite la comprensión sencilla de lo que se ha desarrollado. Un aporte importante es que recomienda utilizar el ancho de 80 columnas para permitir la lectura del código en cualquier pantalla e imprimirlo, de ser necesario, en una fuente legible.

- **Nombres:** La regla principal es el uso de nombres significativos para hacer el código entendible.
 1. La principal regla es que los nombres para identificadores de valores y tipos deben escribirse con letra minúscula y separadas con guiones bajos.
 2. Los constructores deben escribirse con la primera letra en mayúscula y usar minúsculas separadas con guiones bajos.
 3. Los módulos deben escribirse con la primera letra en mayúscula y pueden usar mayúsculas y minúsculas separadas con guiones bajos.
- **Comentarios.**
 1. Comentar solamente cuando haya alguna dificultad.
 2. Evitar los comentarios nocivos y muy largos.
 3. Evitar comentar los cuerpos de las funciones, es mejor realizar un comentario al comienzo de la función.
 4. Los comentarios se definen dentro de los delimitadores: (**Comentario**)
- **Espaciado:** Es necesario siempre separar las palabras con espacios en blanco, nunca debe hacerse usando tabulaciones.

- **Formato:** A continuación, se presentan algunos casos particulares de cómo se debe escribir el código.
 1. El uso de delimitadores va seguido por espacio en blanco. Por ejemplo, cuando se utilizan comas y paréntesis.
 2. Se debe considerar la escritura de funciones con una altura no superior a las 70 líneas esto con el fin de que pueda visualizarse en una pantalla completa.
 3. En los patrones la barra inicial no es necesario colocarla. Y si una definición es muy larga debe romperse a la siguiente línea inmediatamente después de la fecha.

- **Indentación:** La indentación es importante ya que permite una mejor legibilidad del código.
 1. Mantenga siempre el mismo estilo de indentación.
 2. Considere que en OCAML se respeta el ancho de 80 columnas.
 3. Lo recomendable para la indentación son 1 o 2 espacios, nunca usar tabulaciones para este fin.
 4. El cuerpo de una función normalmente se indenta con el nombre de esta y en el cuerpo se utilizan 1 o 2 espacios como mencionamos anteriormente.
 5. Existe el caso especial con los patrones, estos se introducen utilizando una barra vertical. Y las cláusulas de coincidencia del patrón se alinean a un mismo nivel de indentación.