

Documentación

Convenciones

Para describir cada uno de los Tipos de datos se utilizará la siguiente estructura:

Tipo

Nombre del Tipo de dato.

Descripción

Propósito del Tipo. Rol que cumple dentro de la Batalla Espacial.

Valores

Valores posibles para aquellos tipos que son enumerados.

Métodos

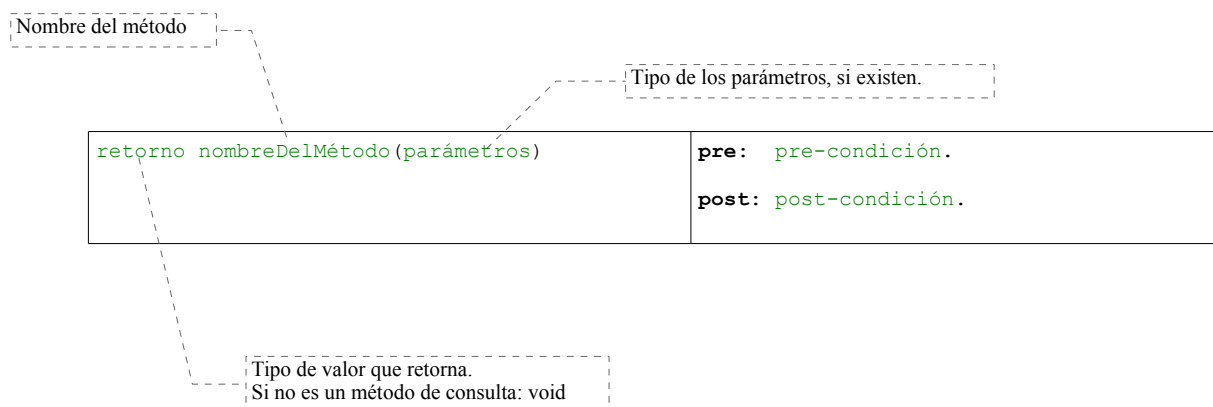
Lista de todos los métodos disponibles para ese Tipo. Se indica el Tipo de valor que retorna, el nombre del método y los Tipos de valores que recibe como parámetro.

Se utiliza la palabra “*void*” para aquellos métodos que no devuelven ningún valor (*void* se puede traducir como vacío)

Además se indican la pre y post-condición de cada método.

La pre-condición (si la tiene) es lo que debo cumplir antes de invocarlo.

La post-condición es lo que se cumple luego de ejecutarlo.



Tipo

BatallaEspacial

Descripción

Es un juego de tablero para programadores donde existen Civilizaciones con Naves que parten de una Base, pueden trasladar Sustancias que encuentren en el espacio y tienen poder de ataque. En el espacio también existen Asteroides que obstruyen los movimientos de las Naves.

Tipo

Nave

Descripción

Nave de la Civilización Humana. Brinda acceso inmediato al control de una Nave pilotada por Humanos.

Métodos

Método	Pre-condición y Post-condición
Monitor obtenerMonitor()	post: devuelve el Monitor de la Nave que permite conocer el estado general del la Nave.
void despegar()	post: ordena el despegue desde la Base.
void avanzarAlNorte()	pre: no existe otra Pieza ocupando el casillero destino. post: mueve la Nave al casillero contiguo en dirección NORTE.
void avanzarAlSur()	pre: no existe otra Pieza ocupando el casillero destino. post: mueve la Nave al casillero contiguo en dirección SUR.
void avanzarAlEste()	pre: no existe otra Pieza ocupando el casillero destino. post: mueve la Nave al casillero contiguo en dirección ESTE.
void avanzarAlOeste()	pre: no existe otra Pieza ocupando el casillero destino. post: mueve la Nave al casillero contiguo en dirección OESTE.
Radar obtenerRadar()	post: devuelve el Radar de la Nave que permite conocer el contexto del Tablero.
void atacarAlNorte()	pre: existe otra Pieza ocupando el casillero destinatario del ataque. post: ataca a la Pieza en el casillero contiguo en dirección NORTE.
void atacarAlSur()	pre: existe otra Pieza ocupando el casillero destinatario del ataque. post: ataca a la Pieza en el casillero contiguo en dirección SUR.
void atacarAlEste()	pre: existe otra Pieza ocupando el casillero destinatario del ataque. post: ataca a la Pieza en el casillero contiguo en dirección ESTE.
void atacarAlOeste()	pre: existe otra Pieza ocupando el casillero destinatario del ataque. post: ataca a la Pieza en el casillero contiguo en dirección OESTE.
void cargarDesdeNorte(Sustancia, int)	pre: existe otra Pieza ocupando el casillero destino con más sustancia que la cantidad indicada y existe capacidad disponible en la bodega de la Nave. post: traslada la cantidad de sustancia desde el casillero contiguo en dirección NORTE hasta la Nave.

<code>void cargarDesdeSur(Sustancia, int)</code>	<p>pre: existe otra Pieza ocupando el casillero destino con más sustancia que la cantidad indicada y existe capacidad disponible en la bodega de la Nave.</p> <p>post: traslada la cantidad de sustancia desde el casillero contiguo en dirección SUR hasta la Nave.</p>
<code>void cargarDesdeEste(Sustancia, int)</code>	<p>pre: existe otra Pieza ocupando el casillero destino con más sustancia que la cantidad indicada y existe capacidad disponible en la bodega de la Nave.</p> <p>post: traslada la cantidad de sustancia desde el casillero contiguo en dirección ESTE hasta la Nave.</p>
<code>void cargarDesdeOeste(Sustancia, int)</code>	<p>pre: existe otra Pieza ocupando el casillero destino con más sustancia que la cantidad indicada y existe capacidad disponible en la bodega de la Nave.</p> <p>post: traslada la cantidad de sustancia desde el casillero contiguo en dirección OESTE hasta la Nave.</p>
<code>void descargarEnNorte(Sustancia, int)</code>	<p>pre: existe otra Pieza ocupando el casillero destino con capacidad disponible.</p> <p>post: traslada la cantidad de sustancia desde la Nave hasta el casillero contiguo en dirección NORTE hasta la Nave.</p>
<code>void descargarEnSur(Sustancia, int)</code>	<p>pre: existe otra Pieza ocupando el casillero destino con capacidad disponible.</p> <p>post: traslada la cantidad de sustancia desde la Nave hasta el casillero contiguo en dirección SUR hasta la Nave.</p>
<code>void descargarEnEste(Sustancia, int)</code>	<p>pre: existe otra Pieza ocupando el casillero destino con capacidad disponible.</p> <p>post: traslada la cantidad de sustancia desde la Nave hasta el casillero contiguo en dirección ESTE hasta la Nave.</p>
<code>void descargarEnOeste(Sustancia, int)</code>	<p>pre: existe otra Pieza ocupando el casillero destino con capacidad disponible.</p> <p>post: traslada la cantidad de sustancia desde la Nave hasta el casillero contiguo en dirección OESTE hasta la Nave.</p>

Tipo

Sustancia

Descripción

Sustancias capaces de ser transportadas por Naves, Contenedores y Bases.

Valores

Sustancia.ANTIMATERIA

Sustancia.CRISTAL

Sustancia.METAL

Tipo

Monitor

Descripción

El Monitor se encuentra asociado a una Nave de la cual reporta su condición.

Métodos

Método	Pre-condición y Post-condición
<code>int consultarCantidadDeTorpedos()</code>	post: devuelve la cantidad de Torpedos de Fotones que posee disponibles la Nave.
<code>int consultarCarga(Sustancia)</code>	post: devuelve la cantidad de Sustancia sustancia que existe en la bodega de la Nave.
<code>int consultarNivelDeCarga()</code>	post: devuelve un valor entre 0 y 100 que representa el porcentaje de la bodega que es ocupado por la carga. El valor 0 indica que no existe carga, bodega vacía. El valor 100 indica que la bodega está completa.
<code>int consultarNivelDeEscudos()</code>	post: devuelve un valor entre 0 y 100 que representa el nivel de energía que tienen los Escudos [%]. El valor 100 indica máximo poder en los Escudos. El valor 0 indica Escudos fuera de funcionamiento.

Tipo

`Espectro`

Descripción

El Espectro indica el resultado de haber realizar un escaneo con el Radar sobre un casillero.

Indica la presencia o ausencia de algún tipo de Pieza en Tablero.

Valores:

`Espectro.VACIO`

`Espectro.DESCONOCIDO`

`Espectro.ASTEROIDE`

`Espectro.CONTENEDOR`

`Espectro.NAVE`

`Espectro.BASE`

Tipo

Radar

Descripción

El Radar es la herramienta que provee una Nave para conocer el contexto inmediato de la misma.

Métodos

Método	Pre-condición y Post-condición
Espectro escanearNorte()	post: escanea el casillero contiguo en dirección NORTE y devuelve el Espectro encontrado.
Espectro escanearSur()	post: escanea el casillero contiguo en dirección SUR y devuelve el Espectro encontrado.
Espectro escanearEste()	post: escanea el casillero contiguo en dirección ESTE y devuelve el Espectro encontrado.
Espectro escanearOeste()	post: escanea el casillero contiguo en dirección OESTE y devuelve el Espectro encontrado.
int buscarAlNorte(Sustancia)	post: busca en el casillero contiguo en dirección NORTE y devuelve la cantidad de sustancia encontrada.
int buscarAlSur(Sustancia)	post: busca en el casillero contiguo en dirección SUR y devuelve la cantidad de sustancia encontrada.
int buscarAlEste(Sustancia)	post: busca en el casillero contiguo en dirección ESTE y devuelve la cantidad de sustancia encontrada.
int buscarAlOeste(Sustancia)	post: busca en el casillero contiguo en dirección OESTE y devuelve la cantidad de sustancia encontrada.