ESAAD

ANNEXES EVALUATION POO – TypeScript

*Ne rien écrire sur cette feuille.*

10)

import { Application } from "pixi.js";

import { Vaisseau } from "./objects/Vaisseau";

export class Main extends Application {

constructor() {

super();

const vaisseau = new Vaisseau();

this.addChild(vaisseau);

this.ticker.add(this.update, this);

}

public update() {

console.log(vaisseau);

}

}

new Main();

12)

import { Application } from "pixi.js";

import { Vaisseau } from "./objects/Vaisseau";

export class Main extends Application {

constructor() {

super();

const vaisseau = new Vaisseau();

this.addChild(vaisseau);

console.log(vaisseau.missiles);

}

}

new Main();

import { Sprite } from "pixi.js";

import { Missile } from “./Missile”

export class Vaisseau extends Sprite {

constructor() {

super();

let missiles:Missile[] = [];

}

}

16)

import { Application, Sprite } from "pixi.js";

export class Main extends Application {

private \_image = Sprite.from("bunny.png");

constructor() {

super();

this.addChild(this.\_image);

this.\_image.interactive = true;

this.\_image.on("pointerdown", this.\_onDown.bind(this));

this.ticker.add(this.update, this);

}

public update() {

this.\_image.rotation += 0.1;

}

private \_onDown() {

console.log("image clicked");

}

}

new Main();

19)  
import { Application } from "pixi.js";

import { Vaisseau } from "./objects/Vaisseau";

export class Main extends Application {

constructor() {

super();

this.addChild(this.vaisseau);

this.vaisseau = new Vaisseau();

this.addEventListener("keydown", keyCallback);

}

private keyCallback(keyData) {

if (keyData.keyCode == 38)

this.vaisseau.y -= 5;

}

}

new Main();

20)

import { Application } from "pixi.js";

import { Vaisseau } from "./objects/Vaisseau";

export class Main extends Application {

constructor() {

super();

for (let i = 0; i < 10; ++i) {

let vaisseau = new Vaisseau();

vaisseau.x = 200 \* Math.random();

this.addChild(vaisseau);

}

this.interactive = true;

this.addListener('mousedown', mouseDownCallback.bind(this));

}

private mouseDownCallback(mouseData) {

}

}

new Main();