

Chapeau
<pre> # nomString: String # p1: PointPlan # p2: PontPlan # p3: PointPlan # ensSegments: ArrayList<Segment> </pre>
<pre> + Chapeau() + Chapeau(nomString: String, p1: PointPlan, p2: PointPlan, p3: PointPlan) + Chapeau(c: Chapeau) + getNomChap(): String + getp1(): PointPlan + getp2(): PointPlan + getp3(): PointPlan + getChap(): ArrayList<Segment> + setNomChap(nom: String) + setp1(p1: PointPlan) + setp2(p2: PointPlan) + setp3(p3: PointPlan) + SetChap(ensSegments: ArrayList<Segment>) + deplacer(arg0: int, arg1: int): void + dessiner():ArrayList<Segement> + typeForme(): String </pre>

Triangle
<pre> # nom: String # p1: PointPlan # p2: PontPlan # p3: PointPlan # ensSegments: ArrayList<Segment> </pre>
<pre> + Triangle() + Triangle(nom: String, p1: PointPlan, p2: PointPlan, p3: PointPlan) + Triangle(t: Triangle) + Triangle(t: Triangle) + getNomTrig(): String + getp1(): PointPlan + getp2(): PointPlan + getp3(): PointPlan + getTrig(): ArrayList<Segment> + setNomTrig(nom: String) + setp1(): String + setp2(): String + setp3(): String + setTrig(ensSegments: ArrayList<Segment>) + deplacer(arg0: int, arg1: int): void + dessiner():ArrayList<Segement> + typeForme(): String </pre>

Quadrilatere
nom: String # p1: PointPlan # p2: PontPlan # p3: PointPlan # p4: PointPlan # ensSegments: ArrayList<Segment>
+ Quadrilatere() + Quadrilatere(nom: String, p1: PointPlan, p2: PointPlan) + Quadrilatere(q: Quadrilatere) + getNomQuad(): String + getp1(): PointPlan + getp2(): PointPlan + getQuad(): ArrayList<Segment> + setNomQuad(nom: String) + setp1(p1: PointPlan) + getp2(p2: PointPlan) + setQuad(ensSegments: ArrayList<Segment>) + deplacer(arg0: int, arg1: int): void + dessiner():ArrayList<Segement> + typeForme(): String

Ardoise
WIDTH: int # HEIGHT: int # MIN_X: int # MAX_X: int # MIN_Y: int # MAX_Y: int # STEP: int # NB-POINT: int # formes: ArrayList<Forme>
+ Ardoise() + ajouterForme(f: Forme): void + deplacer(typeForme: String, deplacementX: int, deplacementY: int): void + dessinerGraphique(): void + test(): void

Forme
nomForme: String
+ Forme() + Forme(nomForme: String) + getNomForme() : String + setNomForme(nomForme: String) : void + dessiner() : ArrayList<Segment> + deplacer(deplacementX: int, deplacementY: int) : void + typeForme() : String + toString() : String

PointPlan
abscisse: int # ordonnee: int
+ PointPlan(x: int, y: int) + getAbscisse(): int + setAbscisse(abscisse: int): void + getOrdonnee(): int + setOrdonnee(ordonnee: int): void + toString(): String + deplacer(deplacementX: int, deplacementY: int): void

Segment
pointDepart: PointPlan # pointArrivee: PointPlan
+ Segment(pointDepart: PointPlan, pointArrivee: PointPlan) + getPointDepart(): PointPlan + setPointDepart(pointDepart: PointPlan): void + getPointArrivee(): PointPlan + setPointArrivee(pointArrivee: PointPlan): void + dessiner(): ArrayList<Segment> + deplacer(deplacementX: int, deplacementY: int): void + typeForme(): String + toString(): String

Etoile
<pre> # nom: String # chapeau1: Forme # chapeau2: Forme # chapeau3: Forme # chapeau4: Forme # ensSegments: ArrayList<Segment> </pre>
<pre> + Etoile() + Etoile(nom: String, chapeau1: Forme, chapeau2: Forme, chapeau3: Forme, chapeau4: Forme) + Etoile(e: Etoile) + getNomEtoile(): String + getChapeau1(): Forme + getChapeau2(): Forme + getChapeau3(): Forme + getChapeau4(): Forme + getEtoile(): ArrayList<Segment> + setNomEtoile(nom: String) + setChapeau1(chapeau1: Forme) + setChapeau2(chapeau2: Forme) + setChapeau3(chapeau3: Forme) + setChapeau4(chapeau4: Forme) + setEtoile(ensSegments: ArrayList<Segment>) + deplacer(arg0: int, arg1: int): void + dessiner(): ArrayList<Segment> + typeForme(): String </pre>

Maison
nom: String # quad11: Forme # quad22: Forme # chapeau: Forme # ensSegments: ArrayList<Segment>
+ Maison() + Maison(nom: String, quad1: Forme, quad2: Forme, chapeau: Forme) + Maison(m: Maison) + getNomMaison(): String + getquad1(): Forme + getquad2(): Forme + getChapeau(): Forme + getMaison(): ArrayList<Segment> + setNomMaison(nom: String) + setquad1(quad1: Forme) + setquad2(quad2: Forme) + setchapeau(chapeau: Forme) + setMaison(ensSegments: ArrayList<Segment>) + deplacer(arg0: int, arg1: int): void + dessiner(): ArrayList<Segment> + typeForme(): String