**Patron Composite**

Question #1-

1. L’intention du patron Composite est de traiter les objets individuels, les « leafs » et les objets multiples , les « composite », composés récursivement à partir des objets individuels, de manière uniforme.
2. AJOUTER IMAGE.

Question #2-

La première abstraction est la présence d’une classe abstraite, Objet3DAbs, qui n’a que des méthodes virtuelles pures, et de laquelle dérivent deux autres classes, soit Objet3DComposite et PrimitiveAbs. La responsabilité de cette abstraction est de permettre à un Objet3DComposite d’être composé et de Primitives, et d’autres Objet3DComposite. Ainsi, lorsqu’on veut poser une action sur un Objet3D, par exemple, modifier le point central de l’objet, il est possible d’appeler la méthode appropriée sur tous les objets dans son conteneur, soit les objets qui le composent, sans qu’il n’y aille d’erreur d’exécution dû au fait qu’une méthode soit appelée sur le mauvais objet.

La deuxième abstraction, est la classe abstraite PrimitiveAbs, de laquelle dérivent les trois classes concrètes Cube, Sphere et Cylinder. Cette abstraction permet d’implémenter de manière homogène les méthodes qui sont communes à toutes les primitives, mais de laisser les classes concrètes implémenter les méthodes qui sont spécifiques à leurs attributs. De cette façon, lorsqu’un Objet3DComposite appelle des méthodes sur les objets qui le composent, il n’a pas à se soucier de la classe concrète de ses primitives, la bonne méthode sera toujours appelée.