* AliensMovement:
  definition: charge on the movement of the aliens and arrange them in
      formation. accept list of aliens.
  functions: - moveAll: move all the aliens, move x by dt, check if the movement
            is in the gui, if not - the aliens need to move by y and dt.
       - time passed: call to moveAll.
       - drawOn: don't do nothing.
       - addToGame: add this to the gameLevel object.
       - lastColumn: order the list of the aliens int columns,
            return the last column of the aliens.
       - backToStart: remove all aliens to the start position and initialize
            their speed.
       - shoot: make in random position shot and returns it.
       - makeListOfLists: order the aliens in list of lists.
       - getToSheild: check if the aliens get to the sheild.
       - removeAliens: remove all the aliens from the list.

* Alien:
  definition: charge on the alien type.
  functions: - getRectStart: get the start rect of the alien - the rect int the
          position when the game begin.
         - removeFromGame - remove the alien from the game.
* Shot:
  definition: charge of the shot. make a ball and tell if the shot is from
        alien or from paddle.
  functions: shotFromAlien - return boolean value.

* SpaceInvaders:
  definition:charge to make gamelevel type of space invaders.
  functions: - all of the functions of gameLevel.
        - aliens: make aliens list and set each alien's position,
            and set their image.
        - blocksShield: make the sheild blocks.
        - numberOfBlocksToRemove: 50, the number of the aliens.
        - shieldStart: return the start of the sheild.

how i implemented:
* aliens formation: in SpaceInvaders i created list of aliens (in the function aliens)
            then i ordred the aliens in formation of list of lists in the class
            AliensMovement (in the function makeListOfLists) - each item
            in the list is column.
* the shileds: in SpaceInvaders i created loop inside loop. inside the second loop i
          created small blocks, each block is 5X5. then i checked if the i equals
          to specific value - if true, the value of i "jump" 80 pixels ahead.
* shots by aliens: i created shot object in AliensMovement in the function shoot.
            then in GameLevel i created a double value "dtShootAlien",
            for the dt value. in doOneFrame i decrease the value if
            dtShootAlien with dt. then i checked if the dtShootPaddle is
            <= then 0, if true - make shoot and change the dt to 0.35.
*shots by player: in GameLevel i created a double value "dtShootPaddle",
            for the dt value. in doOneFrame i decrease the value if
            dtShootPaddle with dt. if the user pressed space key, if
            dtShootPaddle <=0 if true - make ball and change the
            dt to 0.35.