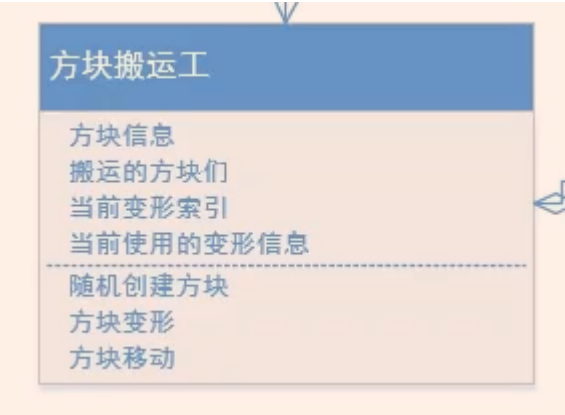


6：搬砖工人随机创建砖块



```

namespace 俄罗斯方块
{
    3 个引用
    class BlockWorker:IDraw
    {
        //方块们
        private List<DrawObject> blocks;
        //选择容器来记录各个方块的形态信息

        //存储方块的几种信息
        private Dictionary<E_DrawType, BlockInfo> blockInfoDic;

        //记录当前随机出来的方块形态信息
        private BlockInfo nowBlockInfo;

        1 个引用
        public BlockWorker()
        {
            //初始化, 砖块信息, 后续使用直接取
            blockInfoDic = new Dictionary<E_DrawType, BlockInfo>()
            {
                {E_DrawType.Cube, new BlockInfo(E_DrawType.Cube) },
                {E_DrawType.Line, new BlockInfo(E_DrawType.Line) },
                {E_DrawType.Tank, new BlockInfo(E_DrawType.Tank) },
                {E_DrawType.Left_Ladder, new BlockInfo(E_DrawType.Left_Ladder) },
                {E_DrawType.Right_Ladder, new BlockInfo(E_DrawType.Right_Ladder) },
                {E_DrawType.Left_Long_Ladder, new BlockInfo(E_DrawType.Left_Long_Ladder) },
                {E_DrawType.Right_Long_Ladder, new BlockInfo(E_DrawType.Right_Long_Ladder) },
            };
            //创建一个方块
            RandomCreateBlock();
        }

        /// <summary>
        /// 随机创建一个下落的方块
        /// </summary>
        1 个引用
        public void RandomCreateBlock()
        {
            //随机方块类型
            Random random = new Random();
            E_DrawType type = (E_DrawType)random.Next(1, 8); 有7中类型, 但是围墙不算在内
                                                                所以是1~7

            //每次新建一个砖块, 就是4个小方形
            blocks = new List<DrawObject>()
            {
                new DrawObject(type), 每次出现的方块都是4个小方形组成的, 不管哪一种类型方块
                new DrawObject(type), 都是4个
                new DrawObject(type),
                new DrawObject(type),
            };
            //需要初始化方块位置
            //原点位置, 随机自定义 方块List中的0个就是我们的原点方块
            blocks[0].pos = new Position(24, 5); 暂时设置原点后续会改动
            //其他三个方块的位置
            //需要取出当前方块的形态, 在进行随机取出变形后的方块
            nowBlockInfo = blockInfoDic[type];
            //不能随机0, 4因为Cube类型方块只有一个数组元素, 其他的都有4个数组元素
            int index = random.Next(0, nowBlockInfo.Count); //在当前类型下的方块中, 随机一种变形后的状态
            Position[] nowPos = nowBlockInfo.list[index];

            for (int i = 0; i < nowPos.Length; i++)
            {
                //取出来的pos是独立的点, 需要转换成相对原点的坐标
                blocks[i + 1].pos = nowPos[i] + blocks[0].pos;
            }
        }

        2 个引用
        public void Draw()
        {
            //绘制创建好的砖
            for (int i = 0; i < blocks.Count; i++)
            {
                blocks[i].Draw(); 每一个元素都是一个DrawObject类, 是一个最小单元的方形, draw就是
                                    根据自己的位置绘制在控制台上
            }
        }
    }
}

```

```

namespace 俄罗斯方块
{
    2 个引用
    class GameScene : ISceneUpdate
    {
        Map map;
        BlockWorker worker;
        1 个引用
        public GameScene()
        {
            map = new Map();
            worker = new BlockWorker();
        }

        2 个引用
        public void upData() // 游戏界面 (主要的逻辑处理更新)
        {
            // 地图更新绘制 // 每一帧都绘制一次
            map.Draw();
            // 绘制砖块
            worker.Draw();
        }
    }
}

```

游戏场景添加工具人类

每次运行，都是随机的方块

