# Java 9 多分辨率图像 API

Java 9 定义多分辨率图像 API，开发者可以很容易的操作和展示不同分辨率的图像了。

以下是多分辨率图像的主要操作方法：

**Image getResolutionVariant(double destImageWidth, double destImageHeight)** − 获取特定分辨率的图像变体-表示一张已知分辨率单位为DPI的特定尺寸大小的逻辑图像，并且这张图像是最佳的变体。。

**List<Image> getResolutionVariants()** − 返回可读的分辨率的图像变体列表。

## 实例

import java.io.IOException; import java.net.URL; import java.net.MalformedURLException; import java.util.ArrayList; import java.util.List; import java.awt.Image; import java.awt.image.MultiResolutionImage; import java.awt.image.BaseMultiResolutionImage; import javax.imageio.ImageIO; public class Tester { public static void main(String[] args) throws IOException, MalformedURLException { List<String> imgUrls = List.of("http://www.runoob.com/wp-content/themes/runoob/assets/img/runoob-logo@2x.png", "http://www.runoob.com/wp-content/themes/runoob/assets/img/runoob-logo.png", "http://www.runoob.com/wp-content/themes/runoob/assets/images/qrcode.png"); List<Image> images = new ArrayList<Image>(); for (String url : imgUrls) { images.add(ImageIO.read(new URL(url))); } // 读取所有图片 MultiResolutionImage multiResolutionImage = new BaseMultiResolutionImage(images.toArray(new Image[0])); // 获取图片的所有分辨率 List<Image> variants = multiResolutionImage.getResolutionVariants(); System.out.println("Total number of images: " + variants.size()); for (Image img : variants) { System.out.println(img); } // 根据不同尺寸获取对应的图像分辨率 Image variant1 = multiResolutionImage.getResolutionVariant(156, 45); System.out.printf("\nImage for destination[%d,%d]: [%d,%d]", 156, 45, variant1.getWidth(null), variant1.getHeight(null)); Image variant2 = multiResolutionImage.getResolutionVariant(311, 89); System.out.printf("\nImage for destination[%d,%d]: [%d,%d]", 311, 89, variant2.getWidth(null), variant2.getHeight(null)); Image variant3 = multiResolutionImage.getResolutionVariant(622, 178); System.out.printf("\nImage for destination[%d,%d]: [%d,%d]", 622, 178, variant3.getWidth(null), variant3.getHeight(null)); Image variant4 = multiResolutionImage.getResolutionVariant(300, 300); System.out.printf("\nImage for destination[%d,%d]: [%d,%d]", 300, 300, variant4.getWidth(null), variant4.getHeight(null)); } }