

```
import datetime

class Author:
    def __init__(self, name, email):
        self.name = name
        self.email = email

class Post:
    def __init__(self, title, content, author):
        self.title = title
        self.content = content
        self.author = author
        self.date_posted = datetime.datetime.now()

class Blog:
    def __init__(self, name):
        self.name = name
        self.posts = []

    def add_post(self, post):
        self.posts.append(post)

    def get_latest_posts(self, num_posts=5):
        sorted_posts = sorted(self.posts, key=lambda post: post.date_posted, reverse=True)
        return sorted_posts[:num_posts]

    def get_posts_by_author(self, author_name):
        return [post for post in self.posts if post.author.name == author_name]
```

```
# Creating authors and posts
author1 = Author("Author1", "author1@email.com")
author2 = Author("Author2", "author2@email.com")

post1 = Post("Title1", "Content1", author1)
post2 = Post("Title2", "Content2", author1)
post3 = Post("Title3", "Content3", author2)

# Creating a blog and adding posts to it
blog = Blog("My Blog")
blog.add_post(post1)
blog.add_post(post2)
blog.add_post(post3)

# Getting latest posts
latest_posts = blog.get_latest_posts()
for post in latest_posts:
    print(f"{post.title} by {post.author.name}")

# Getting posts by a specific author
posts_by_author1 = blog.get_posts_by_author("Author1")
for post in posts_by_author1:
    print(f"{post.title} by {post.author.name}")
```